8 Object Recognition and Shape Representation

Syllabus

Alignment, appearance-based methods, invariants, image eigenspaces

Contents

Object Recognition

Using object models that are known a priori, an object recognition system finds objects in the real world from an image of the world. This is an unusually challenging task. Object recognition is effortless and immediate in humans. The problem of object recognition can be described as a labeling problem based on known object models. Formally, the system should assign correct labels to areas, or a group of regions, in an image including one or more objects of interest (and background) and a set of labels matching to a set of models known to the system. Object recognition and segmentation are inextricably linked: segmentation is impossible without at least a partial recognition of objects, and object recognition is impossible without segmentation.

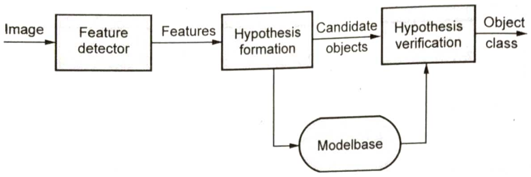

Fig. 8.1 Object representation

Shape Representation

Techniques for representing and describing shapes can be divided into two categories : contour - based methods and region - based methods. The classification is determined by whether shape features are retrieved solely from the contour or from the entire shape region. The many methods are listed under each class. This sub - class is determined by whether the shape is represented in its entirety or in segments/sections (primitives). Based on whether the shape features are taken from the spatial domain or the transformed domain, these approaches can be further divided into space domain and transform domain.

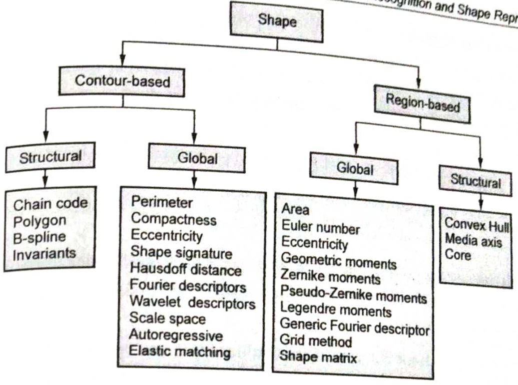

Fig. 8.2 Shape representation classification

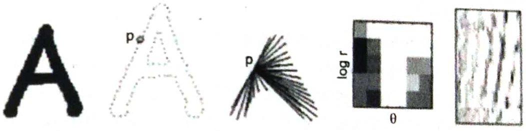

Fig. 8.3 Shape representation example

8.1 Alignment

Task

The major task is to recognize the object's first alignment with an image; the recognition procedure is divided into two parts. The model is aligned with the image in the first stage utilising a modest number of model and image attributes. The alignment is employed in the second stage to convert the model into picture coordinates. When the After determining the model's location and orientation, the model can be directly compared to the image. The alignment operation is based on the discovery that a rigid object's position and orientation may be derived from a small number of position and orientation measures many identification methods, on the other hand, seek out the highest number of consistent model and picture feature pairs with a single rigid object position and orientation. The number of such sets is exponential, necessitating the adoption of diverse strategies to narrow the search.

Aligning a model with an image

Aligning a model with an image requires two pairs of corresponding model and image points, which can be achieved via 2D recognition. Consider the pairs (a_m,a_i) and (b_m,b_i), in which model point a_m is the same as image point b_i. There are three steps to aligning the contours in two dimensions.

The model is first translated till a_m is coincides with a_i and then rotated around the new a_m until the edge $a_m b_m$ coincides with the edge $a_i b_i$.

Finally the scale factor is calculated to align b_m with a_m

These two translations,one rotation and a scale factor make each unoccluded point of the model coincident with its corresponding image point at long on the initial correspondence of (a_m ,a_i) and (b_m ,b_i) is correct.

The alignment approach for 3D from 2D recognition is similar, requiring three pairs of model and picture points to transform and scale the model in three dimensions.

The alignment approach necessitates the discovery of possibly related model and image locations, such as (a_m ,a_i).

The model's probable alignments with the image are then determined using these pairs. To solve for possible alignments, local orientation measurements can also be used. The difficulty of locating alignment points and orientations is addressed. The author describes a system for detecting flat objects with three - dimensional positional freedom.

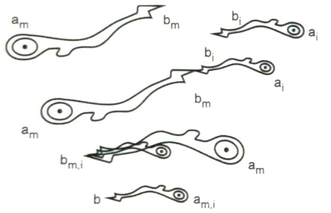

Fig. 8.1.1 2D Translation,Rotation,Scaling of one object to match another

The alignment method of recognition

As we've seen, recognition may be thought of as a search through space for all conceivable positions and orientations of all potential objects. The aim behind the alignment method is to divide their search into two stages. The location, orientations and size of an item are determined in the first stage utilizing only a small amount of data, such as three pairs of model and picture points. The alignment is utilized in the second stage to convert the object model into image coordinates so that it can be compared to the image. An alignment is scored by mapping the model edge into the image with the transformation and comparing the transformed model edges to the image edges. The alignment that maps the most edges onto image edges is the best. By arranging the recognition process as an alignment step followed by a comparison stage, it is turned from an exponential issue of finding the biggest consistent value to a linear one of finding the smallest consistent value. The recognition process is thus converted from an exponential issue of finding the biggest consistent set of model and image points to a polynomial problem of finding the best triplet of model and image points by arranging it as an alignment stage followed by a comparison stage.

Alignment Points

Because the number of possible alignments is cubic in the number of model and image feature pairs, it is crucial to label features differently to limit the number of pairs in the alignment processes. However, while being as distinct as feasible, the label must be reasonably insensitive to partial occlusion, juxta location and projective distortion. If the number of pairs,p_i is kept small, finding the correct alignment requires little or no searching. Zero crossings of curvature, in contrast to curvature maxims, are reasonably robust under projections, fading only when the contour is projected to a straight line. False inflection points exist only when one part of an object partially obscures another. Low curvature regions may produce "inflections" due to extremely slight changes in curvature. Significant inflections are defined as those that occur only in locations where the curvature is not in the range |-e,e|. To segment edge contours, the recognition method described in the next section uses substantial inflection points and low curvature regions.

Recognition Examples

Each sample takes between 2 and 5 minutes to recognize. The model will be able to be aligned with the image using the alignment algorithm. Finally, only planar operations can be used to simulate the alignment approach. Perspective vision has two important practical implications. The first is that objects in the distance appear smaller. The second is that objects that are huge in comparison to the viewing distance appear distorted because distant elements of an object project smaller images than closer components. The main effect of perspective projection for things that are not huge in comparison to the viewing distance is scaling proportionate to the distance from the object. As a result, in these circumstances, perspective projection combined with a linear scale factor is used.

The goal of the alignment challenge is to discover a transformation that maps the plane formed by the three model points onto the picture plane so that each model point corresponds to an image point.

Aligning non - flat objects

It is possible to view a different set of object surfaces from a single three - dimensional model of an object rather than using a single three-dimensional model of an object. A planar model can be matched to a range of distinct photographs of the object using numerous local alignments. A planar perspective of a non - planar object specifies only two - dimensional information about points that have three - dimensional positions in reality. The three model points utilised for alignment are coplanar. A planar model of a three - dimensional object using a single alignment will only match photos taken from near to the same viewing angle as the model view. The first step in this approach is to compute a typical three - point rigid alignment. No further action is taken if the initial alignment does not match any portion of the model to the image. If a match is found, the alignment is utilized to choose more corresponding pairings of model and picture points. Each triangle is aligned separately once the model points are triangulated. This process is repeated until either the model and the image are a good fit or the triangles are small.

8.2 Appearance - Based Methods

Face identification approaches based on appearance have gotten a lot of attention from researchers in fields like biometrics, pattern recognition, computer vision, and machine learning. Despite the fact that people are good at recognizing faces, developing automated face recognition systems remains a difficult task in computer - based automated recognition research. Instead, follow a guideline offered by a psychology research on how people use holistic and local features to achieve a clear and high-level categorization. The proposed papers are divided into the following categories : There are two types of approaches : holistic and hybrid.

Holistic Approaches

Eigenfaces and Fisher faces have proven to be effective in big database tests among appearance - based holistic techniques. Several writers have used entire faces or gabor wavelet filtered full faces as features in holistic techniques.

Eigen images have been one of the key driving forces behind face representation, detection and recognition since the successful low - dimensional reconstruction of faces using KL or PCA projections. It is commonly known that natural photographs contain considerable statistical redundancy. The eigen face approach uses Principal Component

Analysis (PCA), which preserves global structure. For data of all classes, the PCA projects original data into a lower - dimensional subspace spanned by eigen vectors and the associated greatest eigen values of the covariance matrix. The eigen vectors of the collection of faces are called eigen faces.

A prominent facial recognition technique is Linear Discriminant Analysis (LDA), which is based on Fisher Linear Discriminant (FLD). Its goal is to identify the most discriminative features while increasing the determinant of between - class variations to within - class variations ratio. In the field of face recognition, a number of LDA - based algorithms have been presented. However, because of their parametric character, which implies that the data follow a normal distribution, all of these approaches degrade significantly in non - normal distribution instances. For the case of two classes, a non parametric between - class scatter is established and a non parametric technique is proposed to overcome this problem. It does not, however, provide a definition for multi-class problems. A unique approach called Nonparametric Discriminant Analysis (NDA) that extends the definition of the nonparametric between - class scatter matrix to the multi-class problem was used to apply it to face recognition, which is a typical multi - class recognition problem. For the case of two classes, a non parametric between - class scatter is established, and a non parametric technique is proposed to overcome this problem. It does not, however, provide a definition for multi-class problems. To apply it to face recognition, which is a common multi - class recognition problem, researchers developed a new approach called Nonparametric Discriminant Analysis (NDA), which extends the definition of discriminant analysis.

Present a unique method termed Principal Nonparametric Subspace Analysis (PNSA) for extracting nonparametric discriminating features from the principal subspace of a within - class scatter matrix. This will result in the stabilization of the transformation and hence improve recognition performance. To minimize the dimensionality of the proposed descriptor while increasing its discriminative ability, we used a Block-based Fisher's Linear Discriminant (BFLD). Finally, for face recognition, employ BFLD to combine local Gabor magnitude and phase patterns. When magnitude and phase are combined with BFLD and encoded by local patterns, the recognition accuracy improves even further. Due to the singularity problem of the scatter matrix, the traditional LDA cannot be used directly. To address this issue, several extensions of LDA have been proposed in recent years, including pseudo-inverse LDA, Direct LDA and LDA/QR. Face Identification using Principal Component Analysis (IPCA) has improved. The eigenspace is initially constructed using eigen values and eigen vectors. The Eigen faces are produced from this space and the most relevant eigen faces have been chosen using IPCA. Unlike PCA, the input images are categorised using these eigen faces based on euclidian distance. The suggested IPCA approach outperforms the existing methods in terms of accuracy and consistency, as evidenced by the findings. Even with a modest number of training images, the recognition rate is higher, demonstrating an improvement over earlier methods. Kernel PCA (KPCA) is capable of capturing nonlinear relationships between data points and has outperformed standard PCA in several circumstances. The closest neighbor classifier using euclidean distance is used to suggest a method of feature extraction for facial recognition based on KPCA. The results of the experiments demonstrate that KPCA has a high recognition rate. Kernel principal component analysis is a technique for extracting non - linear features.

Hybrid Approaches

Both holistic and local aspects are used in hybrid techniques. The modular eigenfaces technique, for example, employs both global and local eigen characteristics. Eigenfaces can be extended to include eigen features such as eigen eyes, eigen mouths, and so on. Recognition performance as a function of the number of eigenvectors was assessed for eigenfaces only and for the combined representation using a limited set of images (45 people, two views per person, with varied facial emotions such as neutral vs. happy). The Eigen features outperformed the eigenfaces in lower-order spaces.

The Contour's directionality attributes allow transform to extract additional discriminant features in order to improve the effectiveness of the well-known PCA approach when used for face recognition. In order to acquire greater recognition rates by extracting discriminant features The results of the trial show that the contourlet transform outperforms the original PCA approach by a large margin. Furthermore, when paired with PCA, it produces far better classification results than most existing and similar approaches, including the Directional Filter Bank with PCA, Gabor Filter Bank with PCA, and Contourlet with PCA.

Linear feature analysis it has been suggested that a mix of PCA and LFA should be used in real systems. LFA is an intriguing biologically inspired feature analysis tool. Using LFA, it appears that it is better to estimate eigen modes / eigenfaces with large eigen values (and hence are more resilient against noise). In LFA, the entire face stimulates a full 2D array of receptors, each of which corresponds to a different part of the face, yet parts of these receptors may be inactive. LFA is used to extract topographic local features from global PCA modes in order to investigate this redundancy. Unlike PCA kernels I which lack topographic information LFA kernels $K(x_i, y)$ have local support at specified grids x_i.

Fusion of PCA and Bayesian

Face recognition in two dimensions was hampered by pose variations, whilst three-dimensional techniques have a high processing cost. This technique, in addition to improving recognition rates, eliminates the risk of misclassification that can arise with typical single - view systems. To improve the robustness, SVM classifiers are utilized instead of minimal distance classifiers, as opposed to typical PCA - based techniques. When compared to typical 2D single-view face recognition systems, experimental results demonstrate that this two-view face recognition system has a higher recognition rate.

Weighted Eigen Face and BPNN

When computing the covariance matrix of a face image, there is an existing eigenface approach that has a big calculating quantity and demands a significant computer storage capability. These are the K-L transform's bottlenecks. Furthermore, the eigenface technique assigns the same weight to every pixel in a single image. This will reduce the importance of valuable information while exaggerating the importance of unimportant information. Compared to classic appearance - based methods, this method has less computational complexity, a greater recognition rate and more robust. Every face recognition method relies heavily on face image normalization. When image blocking is taken into account, the weighted eigenface method normalizes the face picture to 96*96 pixels. By breaking the facial image into sub blocks and reducing its dimension, the computational complexity and time will be minimized. The typical K-L transform's computing complexity is reduced by using the weighted eigenface technique. The rate of recognition improves by assigning different weights to different areas of the human face based on their relative importance in human face recognition. Within - class average face maximizes within - class information while successfully enlarging different - class information. In comparison to typical algorithms, adaptive learning step algorithm changes BP neural network to jump out of local optimization, reduces learning time and speeds up convergence speed.

8.3 Invariants

The term "invariant object recognition" refers to the ability to recognize an object regardless of picture alterations such as viewpoint, lighting, retinal size, background, and so on. Perceptual constancy is the perceptual outcome of invariance, where the perception of a certain object property is unaffected by irrelevant visual alterations. To a large extent, the mechanisms underlying invariant object recognition have remained unknown. This is due to the fact that previous experimental and computational investigations have mostly concentrated on understanding object identification without these changes, as well as the difficulty of the underlying computational challenges.

Representation of invariants, Learning invariants and it's important warning

Several articles discuss how to create object invariance by utilizing or representing the information in the visual image. Using psychophysical tests, Chuang et al. (2012) demonstrate that non - rigid motion gives a hint to the invariance of dynamic objects. Low-level picture statistics, according to groen et al. (2012), can indicate the extent to which natural textures are invariant across samples. They further show that differences in edge statistics predict changes in evoked brain responses to individual images using electroencephalography (EEG). Bart and Hegdé (2012) show that human participants can detect an object using short informative pieces of an image regardless of fluctuations in illumination using psychophysical studies.The strategy was predicated on the premise that a light source's location and brightness are fixed. As a result, shading variations such as highlights on object surfaces were not taken into consideration by the approach.

External cues to object invariance may be offered in a supervised setting (e.g., Bart and Hegdé, 2012). Finding invariance cues in unsupervised contexts is more difficult. One form of clue comes from the fact that even when an object's look changes, it does so gradually. As a result, changes in object appearance over short, selected lengths of space and time tend to be minor, allowing the visual system to infer that the same item is changing appearance. Continuous Transformation (CT) learning is a theoretical strategy for utilizing this contiguity. Many of the articles in this issue describe models for learning object invariance that use one or more of these rules. Depending on the implementation, the VisNet model can incorporate one or both of these methodologies. While it is commonly assumed that neurons in the higher levels of the visual pathway, such as the inferotemporal cortex, represent object invariance, neurons in the lower levels, such as the primary visual cortex or V_1, can also play crucial roles in implementing various aspects of invariance.

First, object invariance is neither complete nor required at the perceptual level and the underlying brain mechanisms do not have to supply perfect invariance. Second, not all invariance is created equal. Depending on the behavioral setting, some types of invariance may be more relevant or valuable to the visual system than others.Third, in order to learn invariance, the visual system does not need to rely on extensive periods of supervised learning. It's feasible that the system will be able to learn or infer invariance on the fly, without receiving any feedback. Fourth, top-down elements like the behavioral context have a big role in object invariance and lack of it. The articles in this issue, which

largely focus on bottom-up processing of invariance information, do not adequately address this. Finally, contemporary research prefers to treat invariance along various individual stimulus properties (e.g., viewpoint, lighting, etc.) independently for practical reasons. However, the visual system can combine invariance across different visual characteristics, as well as multiple sensory modalities.

Shape invariants : Because they represent shapes based on boundary primitives, shape invariants can also be considered a structural approach. Although most other shape representation systems are invariant under similarity transformations (rotation, translation, and scaling), shape invariants claim that they are dependent on viewpoint. As a result, invariant approaches aim to express properties of boundary configurations that remain intact when subjected to a specific set of transformations. The foundation of invariant theory is a set of transformations that can be combined and inverted. The projective group of transformations, which includes all views as a subset, is considered in vision. A mathematical tool for producing invariants is provided by the group approach. The projective transformation causes a change in coordinates, which is referred to as a group action. Lie group theory comes in handy for creating new invariants.

The number of characteristics utilized to define an invariant is frequently used to name it. An order one invariant is defined on a single feature and is known as a unary invariant; an order two invariant is known as a binary invariant and is defined between two features; similarly, ternary invariant, quaternary invariant, and so on. Geometric invariants include cross-ratios, length ratios, distance ratios, angles, areas, triangles, and invariants from coplanar points; algebraic invariants include determinant, eigen values, and trace; and differential invariants include curvature, torsion, and gaussian curvature. In instances where boundaries can be represented by straight lines or algebraic curves, geometric invariants and algebraic invariants are appropriate.They are commonly used for recognizing man-made objects. Differential invariants can be used to express object boundaries that cannot be represented by lines or algebraic curves.

8.4 Image Eigen Spaces

Turk and Pentland proposed a face recognition method in 1991 that relied on dimensionality reduction and linear algebra ideas to recognize faces. Because this method is computationally less expensive and simple to implement, it was employed in a variety of applications at the time, including handwriting recognition, lip-reading, medical image analysis, and so on.

Pearson proposed PCA (Principal Component Analysis) as a dimensionality reduction technique in 1901. It reduces dimensionality and projects a training sample/data into a compact feature space using eigenvalues and eigenvectors. Let's take a closer look at the algorithm (in a face recognition perspective).

Training Algorithm :

Let's Consider the set of m images of dimension N*N (training images)

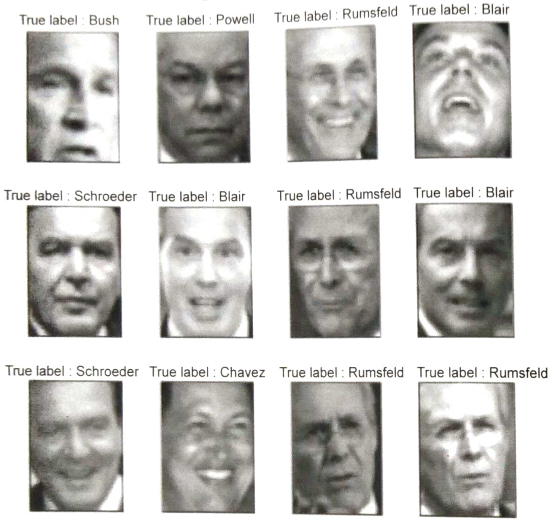

Fig. 8.4.1 Training image with true label

Convert these images into the vectors of size N^2 such that : $x_1, x_2, x_3, x_4, \ldots x_m$

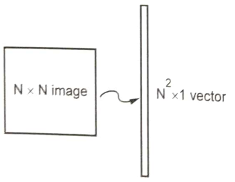

Fig. 8.4.2 Conversion of image with vectors of size N^2

$$\psi = \frac{1}{m} \sum_{i=1}^{m} x_i$$

$$a_i = x_i - \psi$$

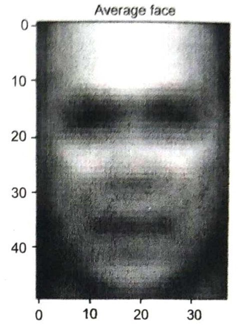

Fig. 8.4.3 Average face

Now we take all face the vectors so that we get a matrix of size of $N^2 * M$.

$$A = [a_1 \, a_2 \, a_3 \ldots a_m]$$

- Now, we find the covariance matrix by multiplying A with A^T. A has the dimensions $N^2 * M$, thus A^T has dimensions $M * N^2$. When we multiplied this gives us the matrix of $N^2 * N^2$, which gives us N^2 eigenvectors of N^2 size which is not computationally efficient to calculate. So we calculate our covariance matrix by multiplying A^T and A. This gives us M * M matrix which has M (assuming $M \ll N^2$) eigenvectors of size M.

$$cov = A^T \cdot A$$

- In this step we calculate eigen values and eigenvectos of above covariance matrix using the formula below,

$$A^T A v_i = \lambda_i v_i$$

$$A A^T A v_i = \lambda_i A \cdot v_i$$

$$C' U_i = \lambda_i u_i$$

where, $$c' = A A^T \text{ and } U_i = Av_i$$

From the above statement it can be concluded that c' and C have same eigenvalues and their eigenvectors are related by the equation $U_i = Av_i$. Thus, the M eigenvalues (and eigenvectors) of covariance matrix gives the **M** largest eigenvalues(and eigenvectors) of c'.

- Now we calculate eigenvector and eigenvalues of this reduced covariance matrix and map them into the c' by using the formula $U_i = Av_i$.
- Now we select the K eigenvectors of *c'* corresponding to the K largest eigen values (where K < M). These eigenvectors has size N^2.
- In this step we used the eigenvectors that we got in previous step. We take the normalized training faces (face – average face) x_i and represent each face vectors in the linear of combination of the best **K** eigenvectors (as shown in the Fig. 8.4.4).

$$x_i - \psi = \sum_{j=1}^{k} w_j u_j$$

These u_j are called **eigenfaces**.

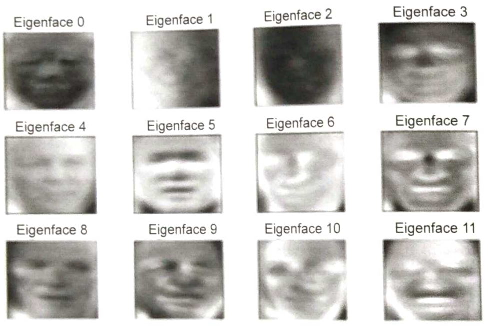

Fig. 8.4.4 Eigen faces

In this step, we take the coefficient of eigenfaces and represent the training faces in the form of a vector of those coefficients.

$$x_i = \begin{pmatrix} w_{1i} \\ w_{2i} \\ w_{3i} \\ \cdot \\ w_{k-i} \end{pmatrix}$$

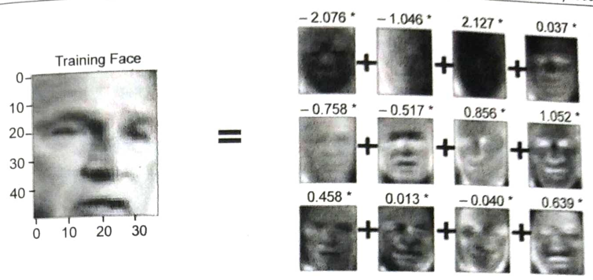

Fig. 8.4.5 Linear combination of eigenfaces

Testing / Detection Algorithm

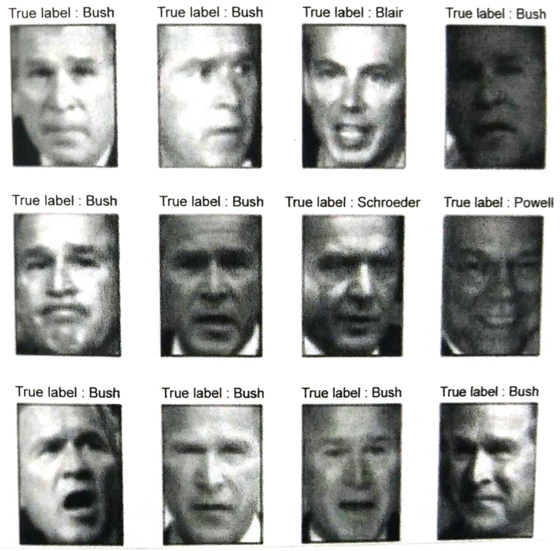

Fig. 8.4.6 Test images with true labels

- Given an unknown face y, we need to first preprocess the face to make it centered in the image and have the same dimensions as the training face.
- Subtract the face from the average face .

$$\phi = y - \psi$$

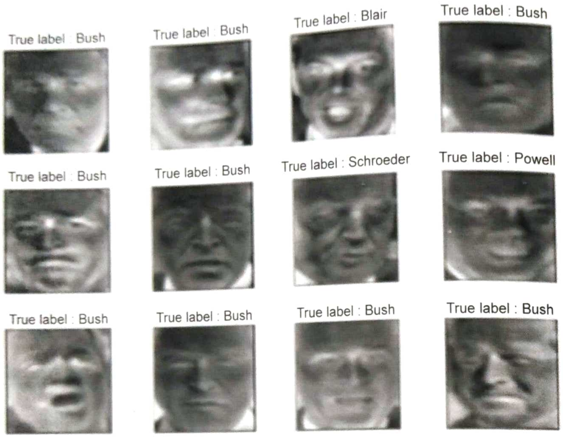

Fig. 8.4.7 Test images - Average images

- Project the normalized vector into eigenspace to obtain the linear combination of eigenfaces.

$$\phi = \sum_{i=1}^{k} w_i u_i$$

- From the above projection, we generate the vector of the coefficient such that

$$\Omega = \begin{pmatrix} w_{1i} \\ w_{2i} \\ w_{3i} \\ . \\ w_{ki} \end{pmatrix}$$

- The vector generated in the above step and subtract it from the training image to get the minimum distance between the training vectors and testing vectors,

$$e_r = \min_j ||\Omega - \Omega_j||$$

- If this e_r is below tolerance level T_r , then it is recognised with *l* face from training image else the face is not matched from any faces in training set.

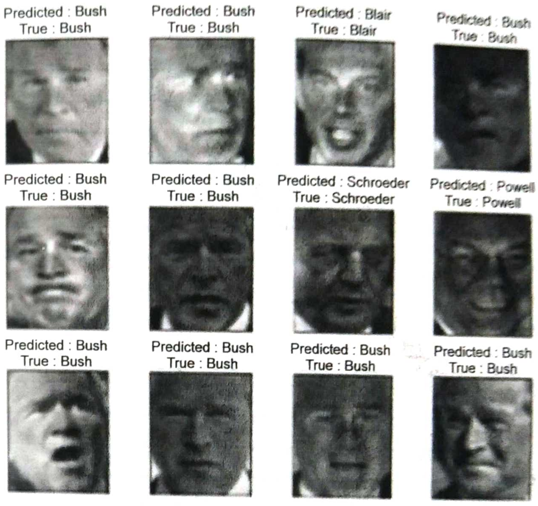

Fig. 8.4.8 Test images with prediction

Advantages :

- Simple to implement and less expensive to compute.
- No prior knowledge of the image (such as a face characteristic) is necessary (except id).

Limitations :

- For training and assessment, a properly centered face is essential.
- The algorithm takes into account lighting, shadows, and the size of the face in the image.
- This method requires a front view of the face to function properly.

□□□