Cơ sở lý thuyết Toán học

- Kỳ vọng
- Phương sai
- Độ lệch chuẩn
- Hiệp phương sai
- Ma trận hiệp phương sai
- Trị riêng và Vector riêng

Giá trị kỳ vọng

- *Ý nghĩa*:
  là giá trị trung bình của một biến ngẫu nhiên. Nó **biểu diễn giá trị trung bình** mà người ta "mong đợi".

- *Cách tính*:
  Trong thống kê, kỳ vọng E_x của một biến ngẫu nhiên X có thể ước lượng bằng trung bình mẫu

$$E_x = \bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

Giá trị kỳ vọng(tt)

- *Ví dụ*: Khảo sát chiều cao của một nhóm 26 học sinh cho kết quả như sau:

Số đo (cm)	155	156	158	159	160	161	162	163	164	165	166	167	168
Số học sinh	1	1	2	3	2	1	4	5	2	1	2	1	1

Kết quả giá trị chiều cao trung bình được tính như sau:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i = \frac{155 \times 1 + 156 \times 1 + \cdots + 168 \times 1}{26} = 161.8$$

Phương sai

- ***Ý nghĩa***:
  Phương sai của một biến ngẫu nhiên là một độ đo sự phân tán của biến đó, nó **hàm ý các giá trị của biến đó thường ở cách giá trị kỳ vọng bao xa.**

- ***Cách tính***:
  phương sai là giá trị kỳ vọng của bình phương của độ lệch của X so với giá trị trung bình của nó:

$$D_x = E((X - E_x)^2)$$

Phương sai(tt)

Kỳ vọng $E_x = \bar{X} = 161.8$; $\boldsymbol{D_x = E((X - E_x)^2)}$

- *<u>Ví dụ</u>*:

-6.8 = 155 - 161.8

Số đo (cm)	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	162	163	164	165	166	167	168
SHS	1	1	2	3	2	1	4	5	2	1	2	1	1
$x_i - \bar{X}$	-6.8	-5.8	-3.8	-2.8	-1.8	-0.8	0.2	1.2	2.2	3.2	4.2	5.2	6.2
$(x_i - \bar{X})^2$	46.24	33.64	14.44	7.84	3.24	0.64	0.04	1.44	4.84	10.24	17.64	27.04	38.44

Phương sai:

$$D_x = \frac{46.24 \text{ x } 1 + 33.64 \text{ x } 1 + 14.44 \text{ x } 2 + \cdots + 38.44 \text{ x } 1}{26}$$

$$= 10.286$$

Độ lệch chuẩn

- *Ý nghĩa*:
  Độ lệch chuẩn là một đại lượng thống kê dùng để **đo mức độ phân tán của dữ liệu so với giá trị kỳ vọng.**
- *Cách tính*:

$$S = \sqrt{D_x}$$

Phương sai và độ lệch chuẩn

- Phương sai: $D_x = E((X - E_x)^2)$
- Độ lệch chuẩn: $S = \sqrt{D_x}$
- *Giống nhau*: S và D_x đều dùng để đánh giá mức độ phân tán của dữ liệu (so với kỳ vọng- số trung bình cộng)
- *Khác nhau*: Khi cần chú ý đến đơn vị đo thì ta dùng độ lệch chuẩn

Ví dụ: Đo chiều cao của học sinh bằng đơn vị $cm \rightarrow$ đơn vị đo của D_x là cm^2 và S là cm

Hiệp phương sai

- *Ý nghĩa*:
  Hiệp phương sai là **độ đo sự biến thiên cùng nhau của hai biến ngẫu nhiên** (khác với phương sai – độ đo sự biến thiên của một biến).
- *Cách tính*:

$$Cov(x,y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{n}$$

- *Ví dụ:*
  kiểu liên hệ hàm số giữa 2 đại lượng biến thiên chiều cao và cân nặng

Hiệp phương sai

- ***Ví dụ***:

Số đo (cm)	155	156	158	159	160	161	162	163	164	165	166	167	168
SHS	1	1	2	3	2	1	4	5	2	1	2	1	1
$x_i - \bar{X}$	-6.8	-5.8	-3.8	-2.8	-1.8	-0.8	0.2	1.2	2.2	3.2	4.2	5.2	6.2

Cân nặng (kg)	40	41	42	43	44	45	46	47	48	49	50	51	52
SHS	1	1	2	3	2	1	4	5	2	1	2	1	1
$y_i - \bar{Y}$	-5.9	-4.9	-3.9	-2.9	-1.9	-0.9	0.1	1.1	2.1	3.1	4.1	5.1	6.1

$\bar{X}$= 161.8 ; $\bar{Y}$= 45.9 ; $Cov(x,y) = 9.7$

Ma trận hiệp phương sai

- ***Định nghĩa*:**
  Ma trận hiệp phương sai của tập hợp m biến ngẫu nhiên là một ma trận vuông (m × m), trong đó :
  - các phần tử nằm trên đường chéo lần lượt là ***phương sai*** tương ứng của các biến này.
  - các phần tử còn lại là các ***hiệp phương sai*** của đôi một hai biến ngẫu nhiên khác nhau trong tập hợp

Ma trận hiệp phương sai

__Ví dụ__: Tập dữ liệu gồm 3 biến ngẫu nhiên (x,y,z)

→Ma trận hiệp phương sai được xây dựng như sau:

$$C(3 \times 3) = \begin{bmatrix} Cov(x,x) & Cov(x,y) & Cov(x,z) \\ Cov(y,x) & Cov(y,y) & Cov(y,z \\ Cov(z,x) & Cov(z,y) & Cov(z,z) \end{bmatrix}$$

Hay

$$C(3 \times 3) = \begin{bmatrix} D_x & Cov(x,y) & Cor(x,z) \\ Cov(y,x) & D_y & Cor(y,z) \\ Cov(z,x) & Cov(z,y) & D_z \end{bmatrix}$$

Ma trận hiệp phương sai(tt)

- *Ý nghĩa*:
  - Nếu ta tính định thức (lấy trị tuyệt đối) của S ta sẽ được giá trị **phương sai tổng quát.** Giá trị này mô tả xu thế phân tán chung của *n* biến ngẫu nhiên quanh các giá trị trung bình của chúng.
  - Nếu ta tính **trị riêng** và **vector riêng** của S , ta sẽ xác định mức độ phân bố của dữ liệu theo các hướng của vector riêng ứng với trị riêng của nó. Hướng nào có trị riêng lớn nhất thì dữ liệu sẽ phân bố theo hướng đó nhiều nhất.

Trị riêng và Vector riêng

- ***Định nghĩa***:
  Cho ma trận $A \in R^n$, Số $\boldsymbol{\lambda} \in R$ được gọi là trị riêng của A nếu tồn tại vector $\boldsymbol{x} = (x_1, x_1, \dots, x_n) \in R^n$ sao cho:

$$A \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix} = \lambda \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_n \end{bmatrix}$$

- Vector X khác 0 gọi là vector riêng của A ứng với giá trị riêng $\boldsymbol{\lambda}$.

Trị riêng và Vector riêng(tt)

- ***Phương pháp***:
  - ***Bước 1:*** *Giải phương trình đặc trưng tìm giá trị riêng:* $\det(A - \boldsymbol{\lambda} I) = 0$

  - ***Bước 2:*** *Tìm vectơ riêng (ứng với giá trị riêng* $\boldsymbol{\lambda}$ *có được ở bước 1) bằng cách giải hệ phương trình tuyến tính thuần nhất :* $(A - \boldsymbol{\lambda}_i I)u = 0$

Trị riêng và Vector riêng(tt)

- ***Ví dụ***: Tìm GTR, VTR của ma trận $\begin{bmatrix} -1 & 3 \\ -2 & 4 \end{bmatrix}$
- ***Bước 1***: Lập phương trình đặc trưng của ma trận A:

$$P(\lambda) = \det(A - \lambda I) = 0 \Leftrightarrow \begin{vmatrix} -1-\lambda & 3 \\ -2 & 4-\lambda \end{vmatrix} = 0 \Leftrightarrow \lambda^2 - 3\lambda + 2 = 0$$

Giải ta được $\lambda_1 = 1, \lambda_2 = 2$

- ***Bước 2***: Tìm các VTR ứng với trị riêng $\lambda_1 = 1$
  Ứng với trị riêng $\lambda_1 = 1$ ta có VTR $u_1 = (x; y)$ là nghiệm của hệ phương trình:

$$(A - \lambda I)u_1 = (A - I)u_1 = 0 \Leftrightarrow \begin{cases} -2x + 3y = 0 \\ -2x + 3y = 0 \end{cases} \Rightarrow 2x = 3y$$

Vậy VTR ứng với GTR $\lambda_1 = 1$ có dạng $u_1 = (3a, 2a) = (3,2)a; a \neq 0$

Tương tự tính VTR ứng với GTR $\lambda_1 = 2$

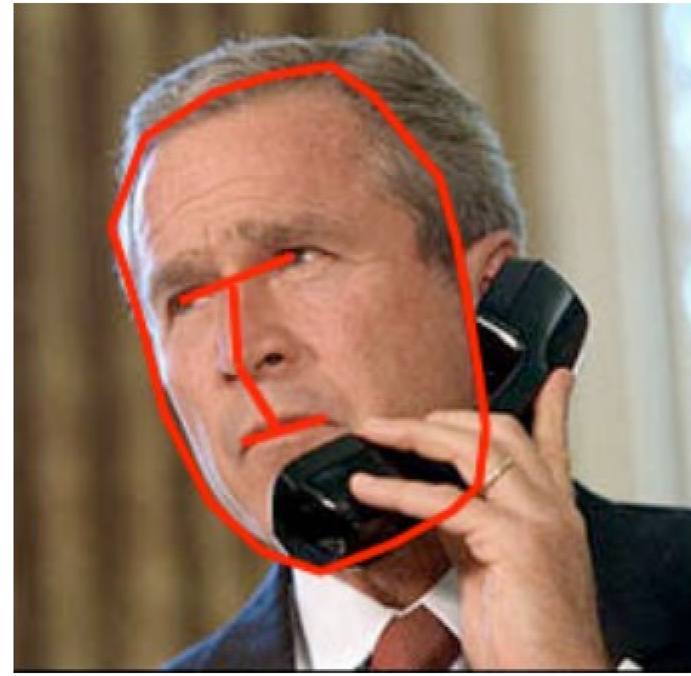
MÁY HỌC

Phần II – PHƯƠNG PHÁP PCA

Nhận dạng khuôn mặt

- Tại sao?
  - Là nơi chủ yếu tập trung sự chú ý trong giao thiệp
  - Thể hiện nhân dạng và xúc cảm
  - Nhiều thay đổi

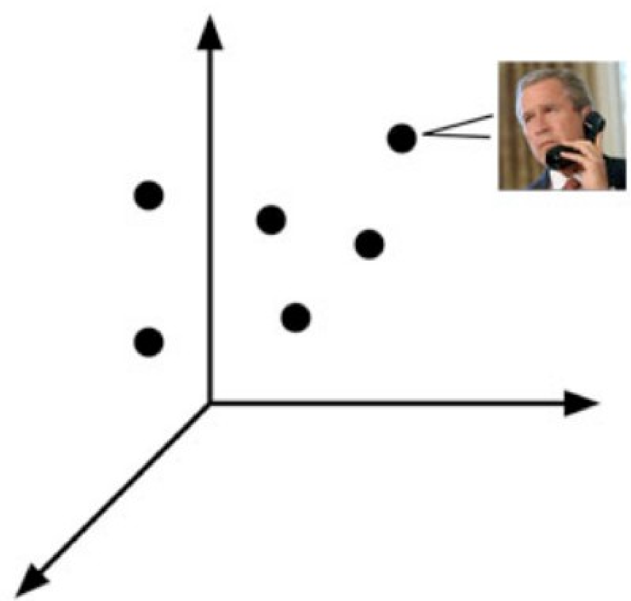
Nhận dạng khuôn mặt



- Các mô hình tính toán
  - Nhận dạng tội phạm
  - Hệ thống an ninh
  - Tương tác giữa người và máy...
- Mục tiêu
  - Nhanh
  - Đơn giản
  - Chính xác trong những môi trường có ràng buộc
- Các đặc trưng cá nhân
  - Mắt, mũi miệng, viền đầu...
  - Mối quan hệ giữa vị trí và kích cỡ

EigenFaces

- Cách tiếp cận eigenface
  - Các ảnh là những điểm trong một không gian vector
  - Dùng PCA để giảm số chiều
  - Face space
    - Sirovich & Kirby 1987
    - Kirby & Sirovich 1990
  - So sánh phép ch ể nhận dạng

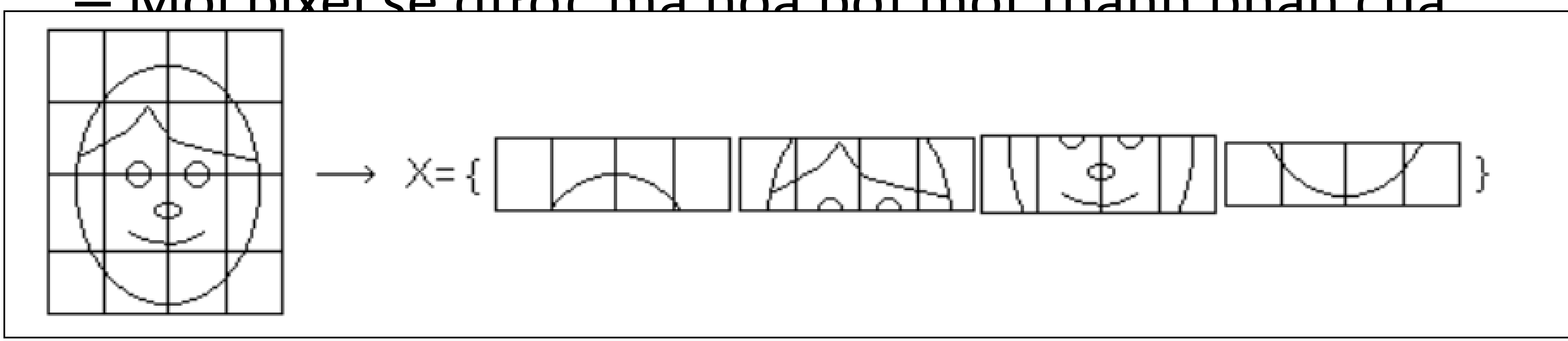


Giới thiệu PCA

- Là một trong những kỹ thuật thành công nhất dùng để nhận dạng và nén ảnh
- Mục tiêu của PCA: giảm số chiều của một tập các vector sao cho vẫn đảm bảo được tối đa thông tin quan trọng nhất của tập học.
- PCA có thể: dự đoán, rút trích đặc trưng, nén dữ liệu,...
- Phù hợp với các ứng dụng có mô hình tuyến tính: xử lý tín hiệu, xử lý ảnh, truyền thông,...
- Được ứng dụng nhiều nhất trong nhận dạng mặt người.

Giới thiệu PCA

- Ứng dụng PCA vào trích chọn vector đặc trưng trong nhận dạng mặt người:
  - Ví dụ: Một face image I(x,y) là 1 mảng 2 chiều NxN, cũng được xem như 1 vector có N2 chiều.
  - Ảnh có size 256x256 → 1 vector 65.536 chiều hay 1 điểm trong không gian 65.536 chiều.
  - Mỗi pixel sẽ được mã hóa bởi một thành phần của



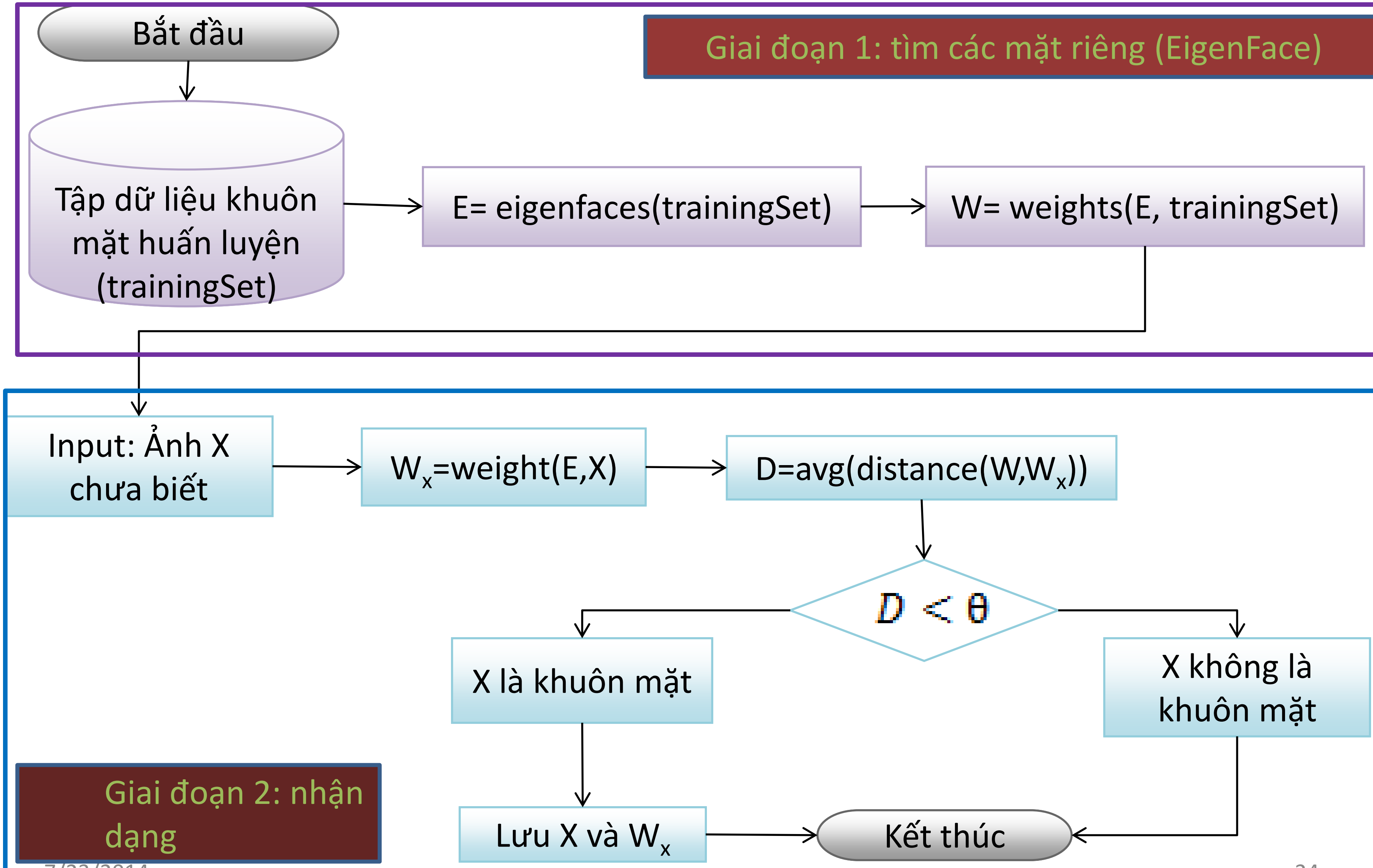
Giới thiệu PCA

- Ứng dụng PCA vào trích chọn vector đặc trưng trong nhận dạng mặt người:
  - ánh xạ 1 vector từ không gian n chiều xuống không gian m chiều (m<n), sẽ đi tìm các trị riêng và vector riêng của ma trận hiệp phương sai C của tập X và giữ lại m vector riêng ứng với m trị riêng lớn nhất làm cơ sở cho không gian m chiều này.
- Dựa trên mô hình của lý thuyết thông tin
  - Phân chia gương mặt người thành một tập nhỏ các ảnh đặc trưng gọi là các mặt riêng (eigenface).
  - Các mặt riêng này được xem như các thành phần chính của tập các ảnh gương mặt ban đầu.

Phương pháp PCA

- Quá trình nhận dạng được thực hiện bằng cách:
  - Chiếu gương mặt mới lên không gian con được định hướng bởi các mặt riêng,
  - Sau đó so sánh nó với vị trí của các ảnh trong tập ban đầu trong không gian mặt riêng. Tìm ra ảnh học gần với ảnh cần nhận dạng nhất.
- Chia thành 2 giai đoạn chính:
  - Giai đoạn tìm các mặt riêng (EigenFace)
  - Giai đoạn nhận dạng

Phương pháp PCA



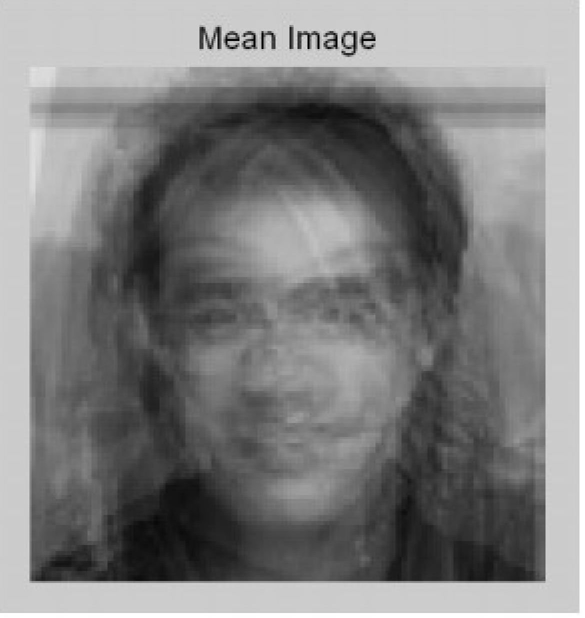
Ví dụ

- Training set

40 faces, 112 x 92 pixels = 10,304 pixels

Ví dụ (tt)

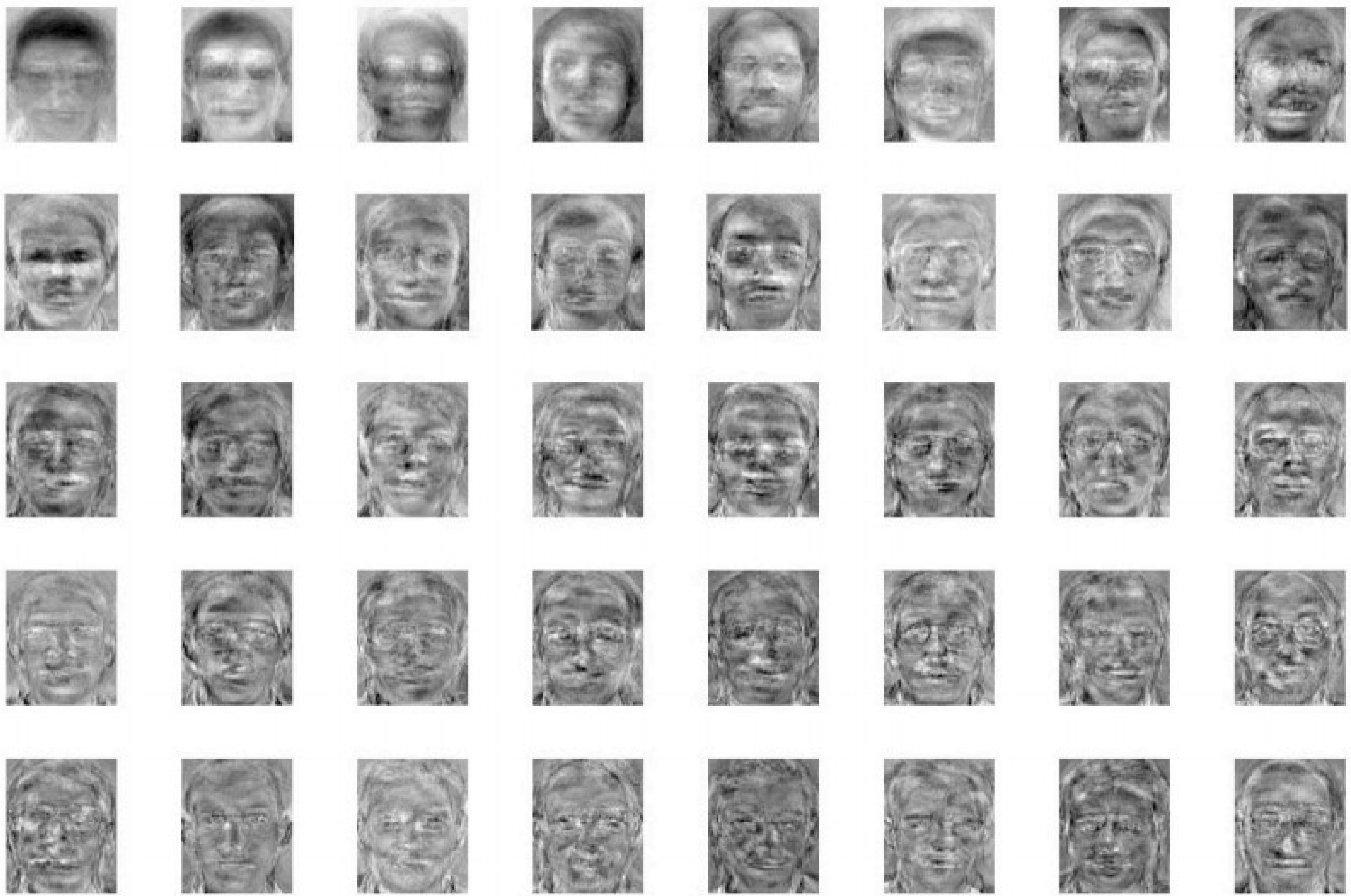
- Ảnh trung bình



Ví dụ (tt)

- Eigengace

X is 10,304 x 40, T is 40 x 40



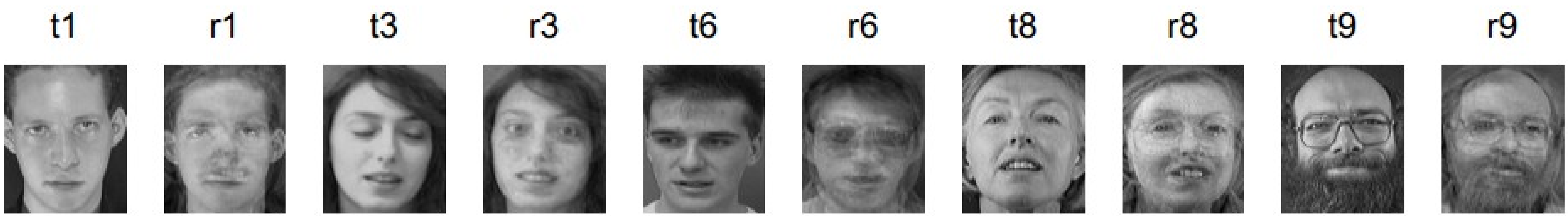
Ví dụ (tt)

- Face space

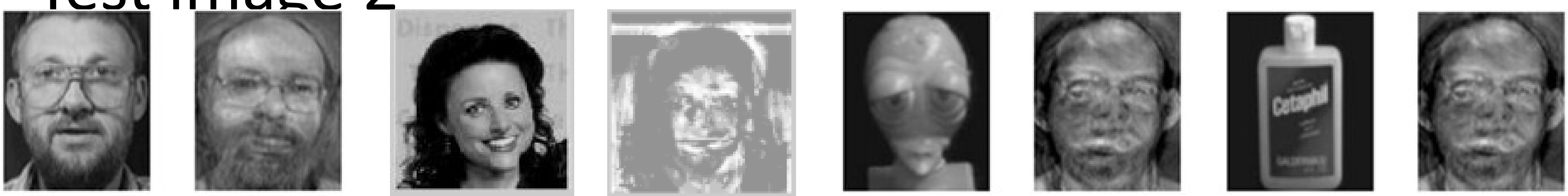
Face Space = top 8 eigenfaces

Ví dụ (tt)

- Test image 1



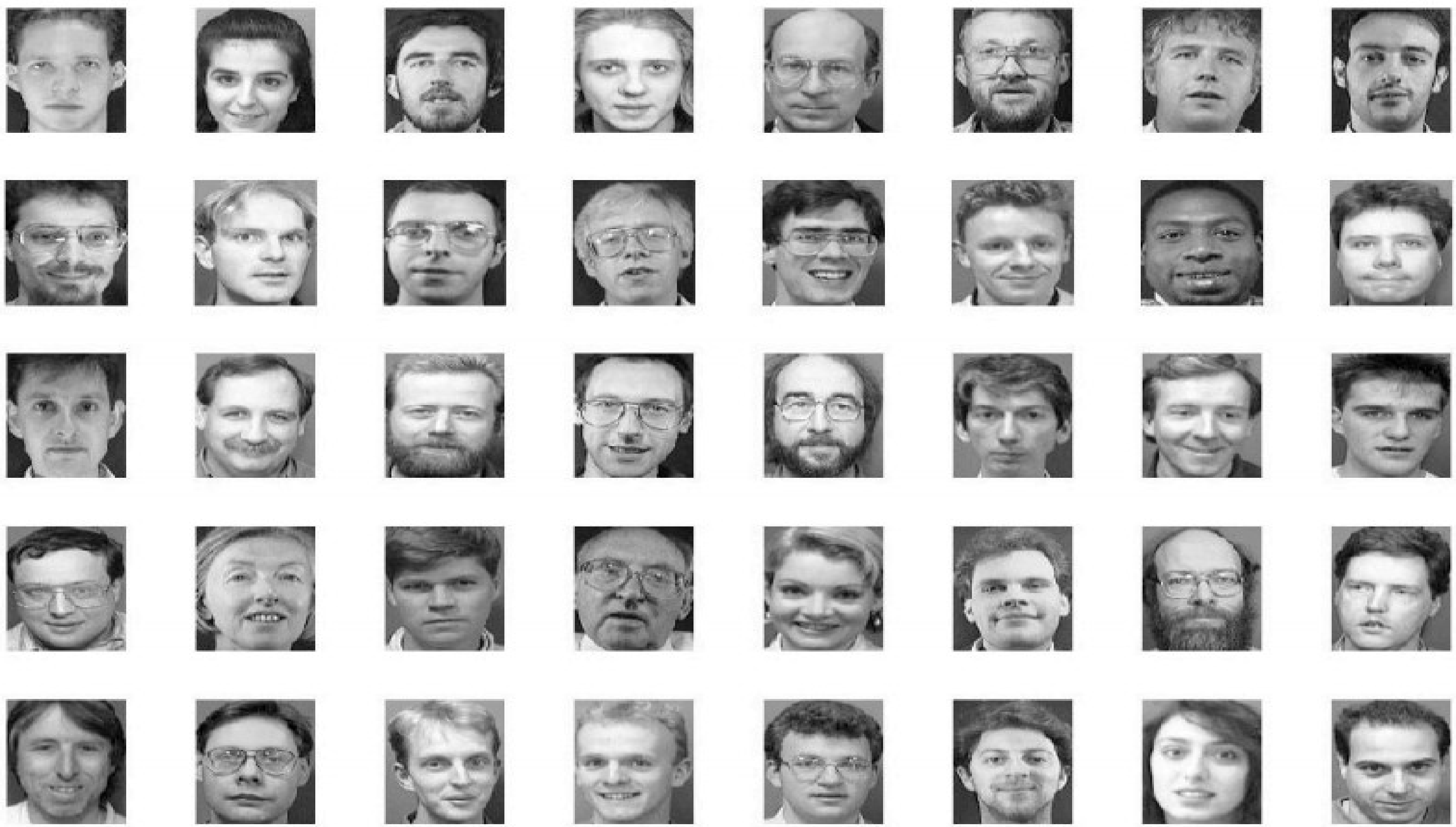
- Test image 2



Giai đoạn 1: Tìm mặt riêng (EigenFace

- Bước 1: Giả sử tập ảnh huấn luyện gồm M ảnh khuôn mặt: $\{\Gamma_1, \Gamma_2, \ldots, \Gamma_M\}$ có kích thước N*N.

40 faces, 112 x 92 pixels = 10,304 pixels

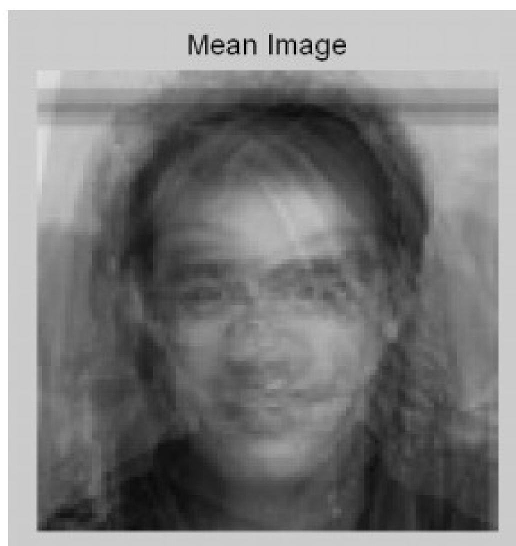


Giai đoạn 1: Tìm mặt riêng (EigenFace

- Bước 2: Tìm ảnh trung bình theo công thức:

$$\Psi = \frac{1}{M}\sum_{M}^{1}\Gamma_i$$

- Với tập ảnh huấn luyện trên ta tính được ảnh trung bình như sau:



Giai đoạn 1: Tìm mặt riêng (EigenFace)

- Bước 3: Tính độ sai khác giữa ảnh huấn luyện Γ_i so với ảnh trung bình Ψ:

$$\Phi_i = \Gamma_i - \psi$$

- Bước 4: Tính ma trận hiệp phương sai C (covariance matrix)

$$C = \frac{1}{M}\sum_{i=1}^{M} \Phi_i \Phi_i^T = A.A^T$$

Với $A = [\Phi_1, \Phi_2, \ldots, \Phi_M]$

$$C = AA^T = \begin{bmatrix} Q_{11} & \cdots & Q_{m1} \\ \vdots & \ddots & \vdots \\ Q_{1n^2} & \cdots & Q_{mn^2} \end{bmatrix}_{n^2 \times m} X \begin{bmatrix} Q_{11} & \cdots & Q_{1n^2} \\ \vdots & \ddots & \vdots \\ Q_{m1} & \cdots & Q_{mn^2} \end{bmatrix}_{m \times n^2}$$

Giai đoạn 1: Tìm mặt riêng (EigenFace

$$C = AA^T = \begin{bmatrix} Q_{11} & \cdots & Q_{m1} \\ \vdots & \ddots & \vdots \\ Q_{1n^2} & \cdots & Q_{mn^2} \end{bmatrix}_{n^2 \times m} X \begin{bmatrix} Q_{11} & \cdots & Q_{1n^2} \\ \vdots & \ddots & \vdots \\ Q_{m1} & \cdots & Q_{mn^2} \end{bmatrix}_{m \times n^2}$$

- Vậy C có kích thước: N^2 x N^2

➡ Vấn đề về tìm vector riêng (eigenvector) u_i của ma trận C khó thực hiện được vì kích thước quá lớn.

- Để tìm eigenvector u_i của C ta thực hiện như sau:

– Giả sử v_i là vector riêng của ma trận A^TA, tức là:

$$A^TAv_i = \mu_i v_i$$

– Nhân 2 vế với ma trận A ta được:

➡ $$AA^TAv_i = \mu_i Av_i$$

Như vậy Av_i là eigenvector của C

Giai đoạn 1: Tìm mặt riêng (EigenFace

- Để tìm eigenvector u_i của C ta thực hiện như sau (tt):
  - Tìm eigenvector và eigenvalue của ma trận L:

$$L = A^T A \qquad Với\ L_{m,n} = \Phi_m^T \Phi_n$$

- Hay:

$$L = A^T A = \begin{bmatrix} Q_{11} & \cdots & Q_{1n^2} \\ \vdots & \ddots & \vdots \\ Q_{m1} & \cdots & Q_{mn^2} \end{bmatrix}_{m*n^2} X \begin{bmatrix} Q_{11} & \cdots & Q_{m1} \\ \vdots & \ddots & \vdots \\ Q_{1n^2} & \cdots & Q_{mn^2} \end{bmatrix}_{n^2*m}$$

  - Khi đó L là ma trận có kích thước là MxM
  - Giả sử v_i là

$$u_i = \sum_{k=1}^{M} v_{ik} \Phi_k \,, i = 1, 2, \dots, M$$

là eigenvector của C hay còn gọi là mặt riêng (eigenface).

Giai đoạn 1: Tìm mặt riêng (EigenFace

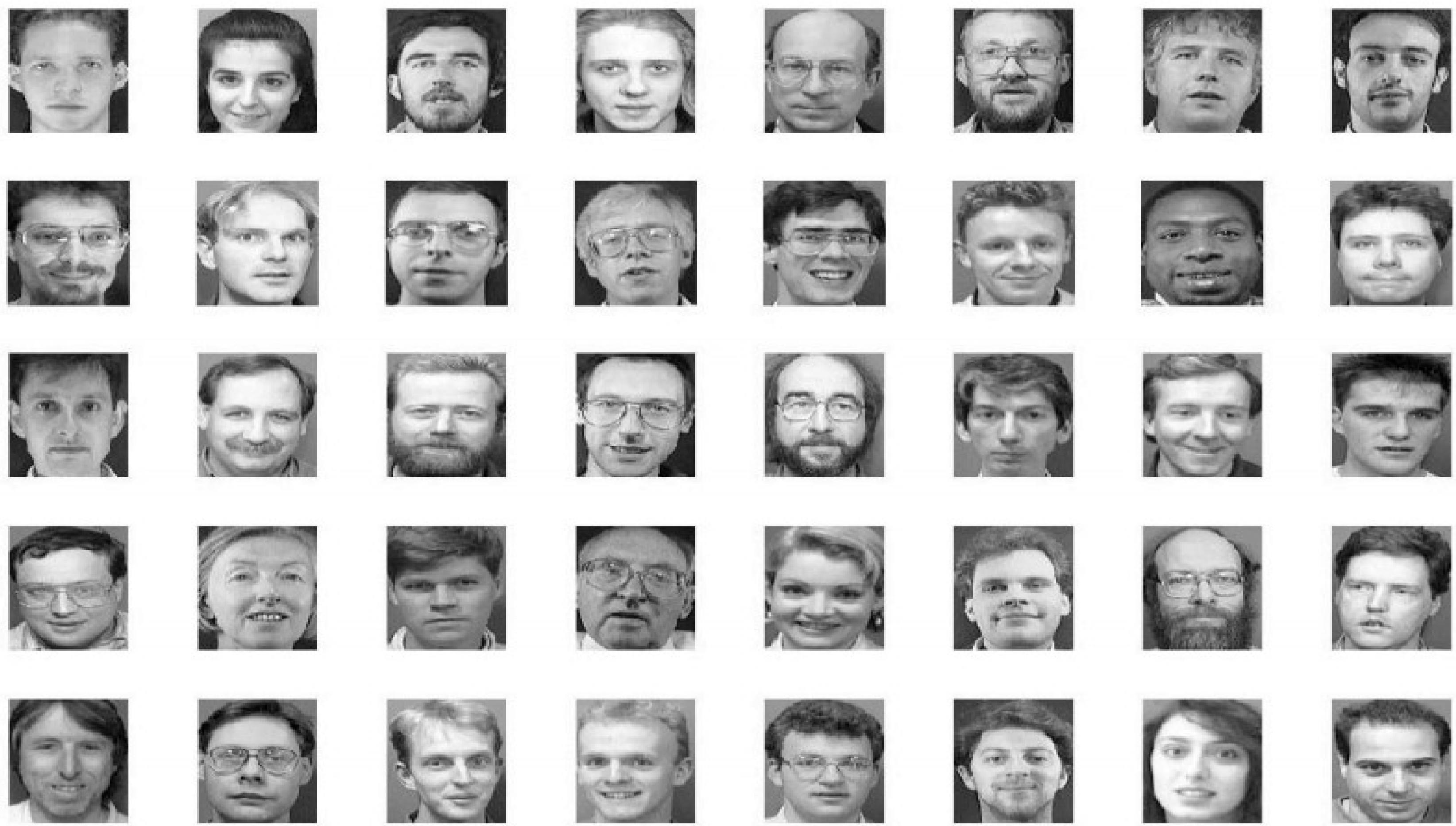
- Từ M eigenvector (u_i), chọn ra M’ eigenvector ứng với M’ giá trị riêng đầu tiên trong mảng các giá trị riêng (được sắp xếp giảm dần).
- Thường chọn M’ sao cho [3]:

$$\frac{\sum_{i=1}^{M'} \mu_i}{\sum_{j=1}^{M} \mu_i} > 0.9 \quad \text{Trong đó } M' < M \ll N^2$$

Giai đoạn 1: Tìm mặt riêng (EigenFace

- M=40

40 faces, 112 x 92 pixels = 10,304 pixels



Giai đoạn 1: Tìm mặt riêng (EigenFace)

- Với ví dụ tập huấn luyện trên tìm được 7 mặt riêng có giá trị riêng lớn nhất (M=40 > M’=8)

Face Space = top 8 eigenfaces

Giai đoạn 2: Nhận dạng

- Bước 1: Với mỗi ảnh huấn luyện $\Gamma_1, \Gamma_2, \ldots, \Gamma_M$, ta chiếu lên không gian mặt M’ chiều:

$$\Omega_k^T = [\omega_1, \omega_2, \ldots, \omega_{M'}] \qquad k = 1, \ldots, M$$

- Với: $\omega_i = u_i^T(\Gamma_k - \Psi), \quad i = 1, 2, \ldots, M'$

Giai đoạn 2: Nhận dạng

- Bước 2: Khuôn mặt mới Γ sẽ được chiếu lên không gian M’ chiều. Kết quả:

$$\Omega^{T} = [\omega_1, \omega_2, \ldots, \omega_{M'}]$$

- Với: $$\omega_i = u_i^T(\Gamma - \Psi), i = 1, 2, \ldots, M'$$

- Bước 3: Tìm mặt thứ k sao cho:

$$\varepsilon_k^2 = \left\| (\Omega - \Omega_k) \right\|^2 \le \theta$$

- Với Ω_k là vector mô tả hay đại diện cho mặt thứ k trong tập huấn luyện, θ là ngưỡng xác định. (khoảng cách Euclide)

Giai đoạn 2: Nhận dạng

- Tuy nhiên, ta cũng cần tính khoảng cách ε của ảnh mới đến face space

$$\varepsilon^2 = \left\| \Phi - \Phi_f \right\|$$

- Với: $\Phi = \Gamma - \psi$ và

$$\Phi_f = \sum_{i=1}^{M'} \omega_i u_i$$

Giai đoạn 2: Nhận dạng

- Có 4 trường hợp khi nhận dạng khuôn mặt mới:
  - Ảnh ở gần không gian mặt và gần 1 lớp ảnh:

    $\varepsilon < \theta$ và $\varepsilon_k < \theta$

    - Known faces
  - Ảnh ở gần không gian mặt và xa tất cả các lớp ảnh

    $\varepsilon < \theta$ và $\varepsilon_k > \theta$

    - Unknown faces
  - Ảnh ở xa không gian mặt và ở gần một lớp ảnh
    - Non-faces
  - Ảnh ở xa không gian mặt và ở xa tất cả các lớp ảnh
    - Non-faces

Giai đoạn 2: Nhận dạng

- Nếu ảnh mới được phân lớp thuộc về lớp k nào đó, ảnh này có thể được thêm vào tập các ảnh ban đầu, và eigenfaces được tính lại.
- Điều này tạo cơ hội để sửa đổi không gian mặt vì hệ thống có thêm các mặt được học.
- Khi ảnh được phân lớp là "unknown" thì nó được dùng để bắt đầu một lớp mặt mới.

Nhận xét

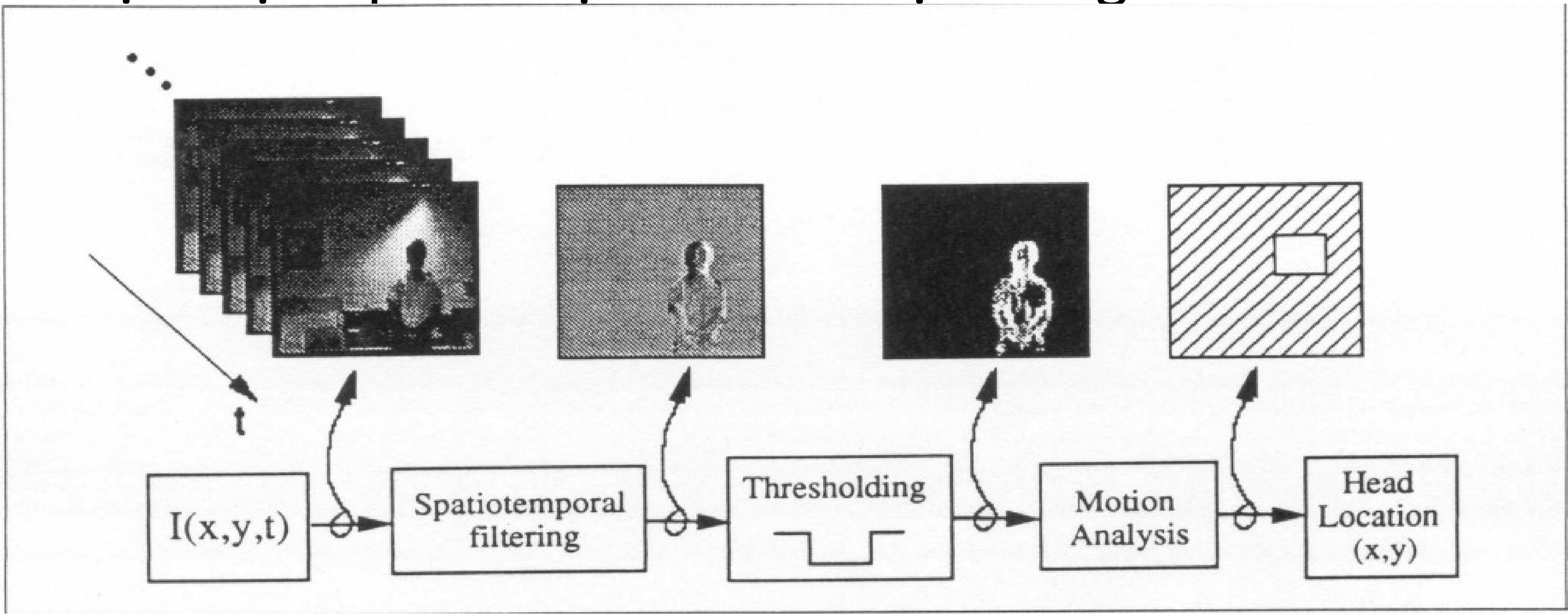
- Các bước xử lý trong phương pháp PCA kết hợp với phương pháp tính khoảng cách Euclides tạo nên một thuật toán nhanh, đơn giản.
- Phương pháp này có ưu điểm:
  - Nhanh
  - Đơn giản
  - Làm việc tốt trong môi trường có ràng buộc về điều kiện (tập đặc trưng nhỏ, góc nghiên bé, ánh sáng…)
  - Không nhạy cảm với những tương đối nhỏ hay từ từ của gương mặt
  - Không phụ thuộc vào mô hình 3 chiều hay các đặc điểm trên khuôn mặt (mắt, mũi, miệng,…)

Nhận xét

- Nhược điểm:
  - PCA phân loại theo chiều phân bố lớn nhất của tập vector. Tuy nhiên, chiều phân bố lớn nhất không phải lúc nào cũng mang lại hiệu quả tốt nhất cho bài toán nhận dạng.
  - PCA rất nhạy với nhiễu.

Mở rộng

- Định vị và phát hiện khuôn mặt trong ảnh và video



- Nhận dạng các khuôn mặt mới

Những vấn đề khác

- Khử nền (Eliminating the background)
  - Trong thực tế, nền có ảnh hưởng đáng kể đến việc nhận dạng
  - Nhân ảnh đầu vào với “cửa sổ” gaussian 2 chiều trên khuôn mặt.
- Tỉ lệ (head size) và hướng không thay đổi
- Phân bố trong face space
- Nhiều góc nhìn