

Eigenfaces Recognition

Hassan AIT BRIK - M2 ISSI - 2015/2016

Université Pierre et Marie Curie
5AI01
Perception et modélisation de
l'interaction

Introduction

Introduction

- ❖ Matthew Turk et Alex Pentland (Massachusetts Institute of technology) - 1991
- ❖ Développement d'un système informatique en temps quasi réel
- ❖ Localisation et suivi par reconnaissance de visage

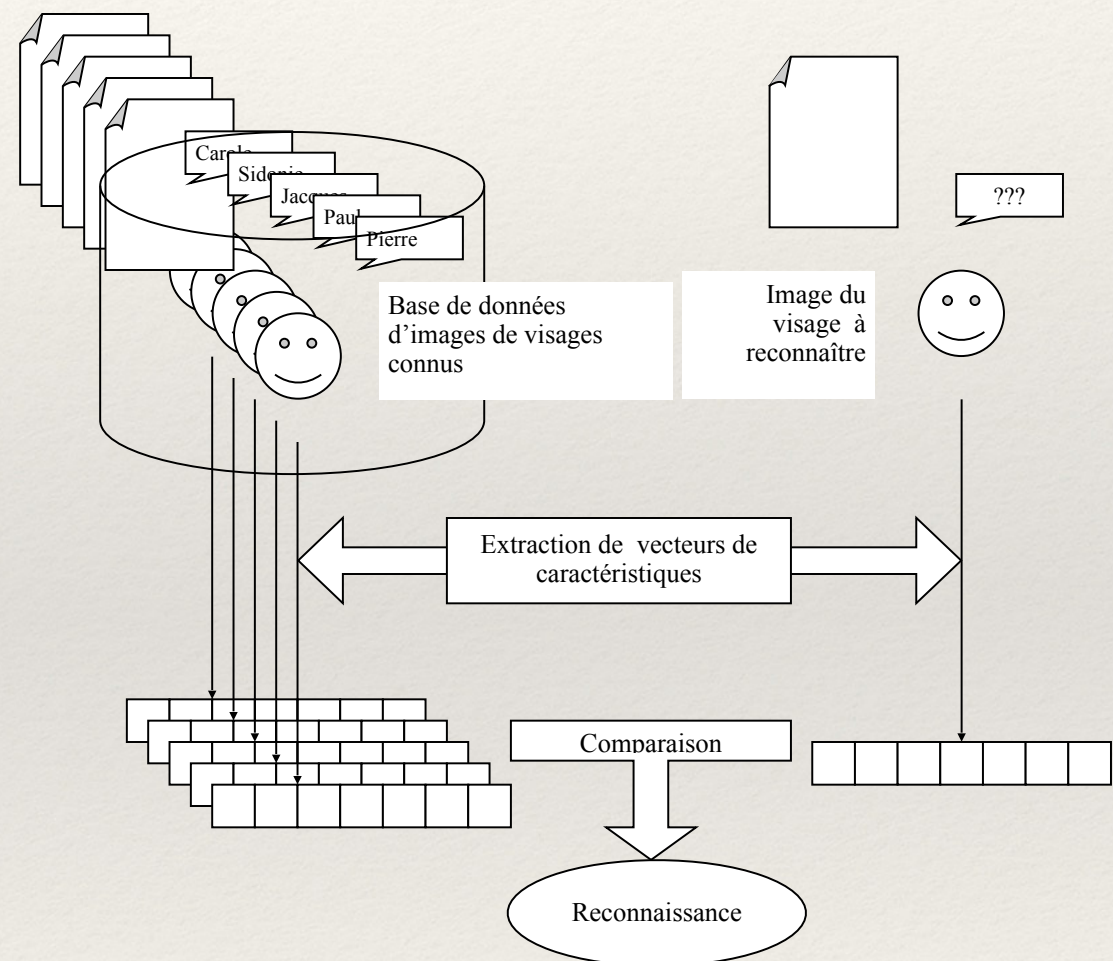
Plan

- ❖ Reconnaissance
 - ❖ Principe
 - ❖ Analyse en Composantes Principales (ACP)
 - ❖ Classification
- ❖ Localisation et détection
 - ❖ Détection de mouvement
 - ❖ Face-map

Reconnaissance

Principe

- ❖ ACP, calcul des visages propres (Eigen-Faces)
- ❖ Comparaison des projections dans l'espace de visage (Face space) par distance euclidienne



Principe



❖ Base de données équilibrée de 110 visages de NbClass = 10 personnes différentes :

- ❖ 70 pour l'apprentissage (66%)
- ❖ 40 pour la classification (34%)

Pour chacune des personnes nous disposons au total de 11 images :

- expressions différentes
- changement de luminosité
- port de lunettes

Analyse en Composantes Principales

On dispose d'une base d'apprentissage de $M = 70$ visages ($L = 7$ visages / personne) de dimension 64×64 ($N = 4096$)

On représente chaque image comme étant un vecteur de N dimensions:

$$\Gamma = [\Gamma_1, \Gamma_2, \dots, \Gamma_M]_{N \times M}$$

On pré-traite les données :

$$\Phi_i = \Gamma_i - \Psi \quad \Psi = \frac{1}{M} \sum_{j=1}^M \Gamma_j$$

Et on calcule :

$$A = [\Phi_1, \Phi_2, \dots, \Phi_M]_{N \times M}$$

Analyse en Composantes Principales

Ensuite, on calcule la matrice de covariance:

$$C = \frac{1}{M} \sum_{k=1}^M \Phi_k \Phi_k^T \quad N \times N$$

Ce qui revient à calculer :

$$C = \frac{1}{M} (AA^T) \cong (AA^T)_{N \times N}$$

Enfin, on calcule les N valeurs propres et les N vecteurs propres de cette matrice.
(Fonction *eig()* sur Matlab)

Analyse en Composantes Principales

Chacun de ces vecteurs étant de dimension N , il est possible de les remettre sous la forme d'une matrice 64×64 .

Les 3 premiers Eigen-Faces:



u_1



u_2



u_3

Analyse en Composantes Principales

Les M exemples s'expriment comme une combinaison linéaire des Eigen-Faces plus le visage moyen:

$$\Phi_i = \omega_{i1}u_1 + \omega_{i2}u_2 + \dots + \omega_{ik}u_k$$

$$\Gamma_i = \Phi_i + \Psi$$

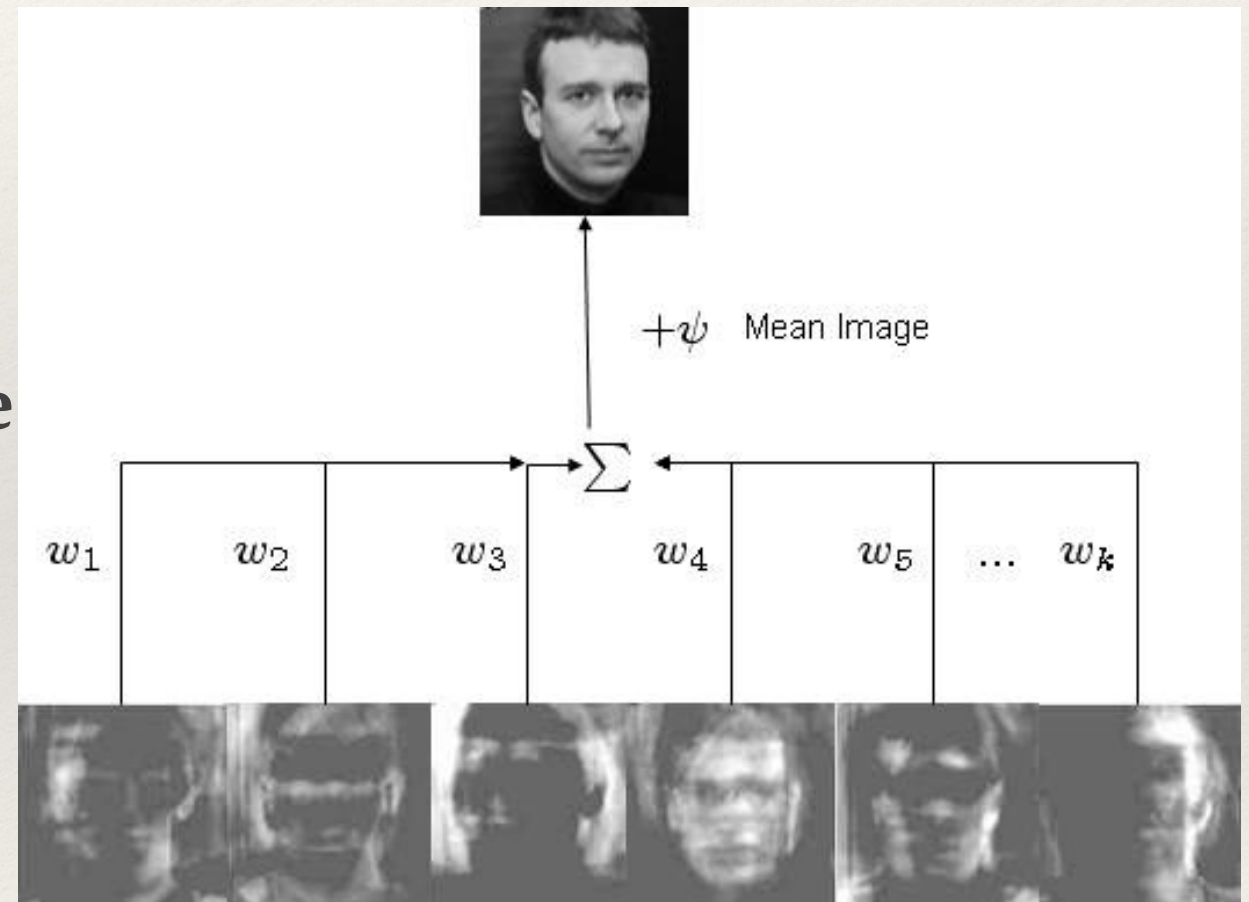
$$\omega_{i1} = \Phi_i^T \cdot u_1 (\text{scalaire})$$

L'espace u_1, u_2, \dots, u_k se nomme **Espace de visage** (Face-space)

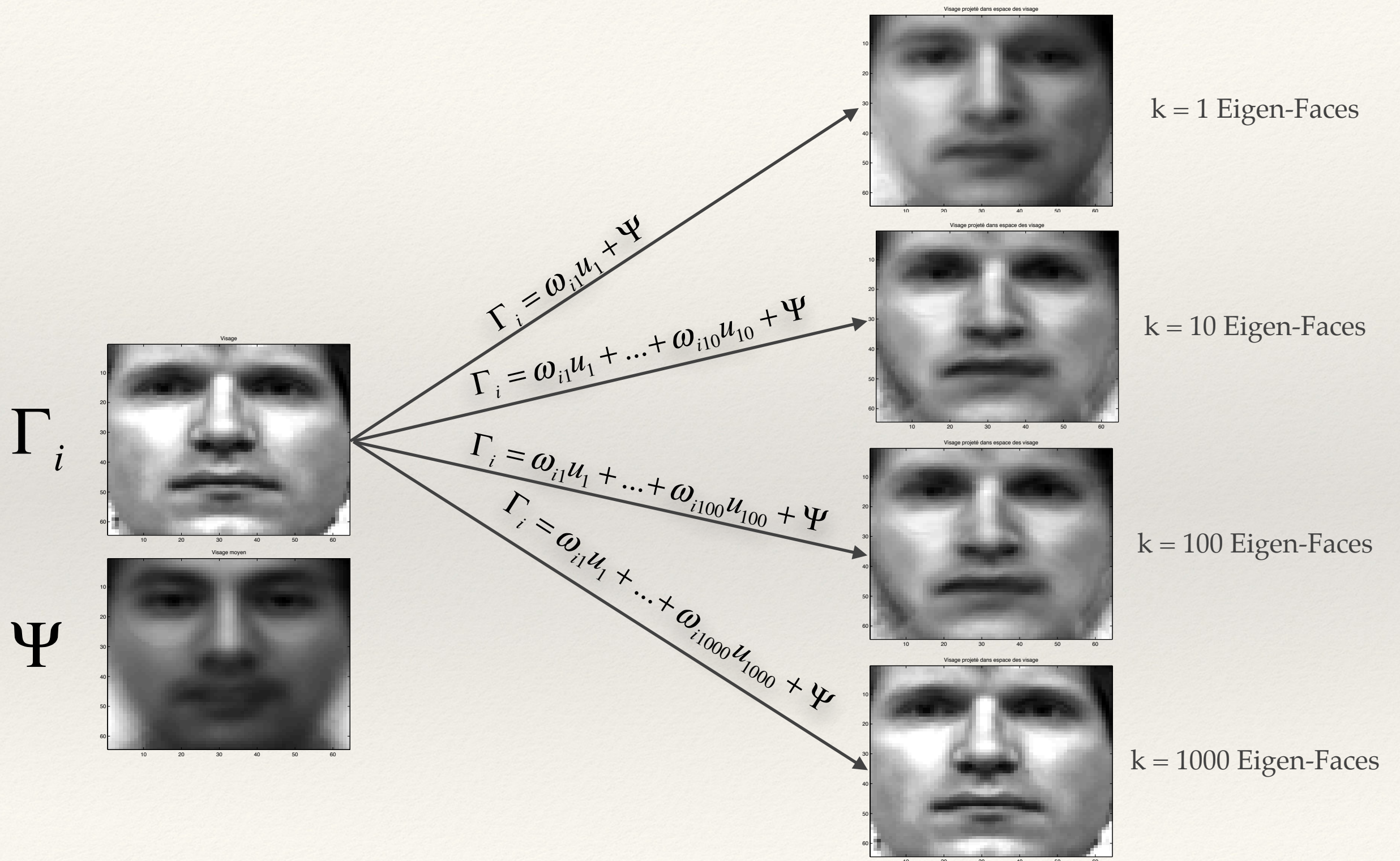
Trouver le bon k ?

Calcule de l'inertie :

$$\frac{\lambda_1 + \lambda_2 + \dots + \lambda_k}{\lambda_1 + \lambda_2 + \lambda_3 + \dots + \lambda_N} \geq 0.95$$



Analyse en Composantes Principales



Classification

❖ Calcule de 2 distances :

❖ Distance par rapport à la classe k

$$\varepsilon_k = d(\Gamma_i, C_k) = \left\| \Gamma_i^{proj} - \frac{1}{L} \sum_{j=1}^L \Gamma_{j,C_k}^{proj} \right\|$$

Projection moyenne de la classe k

❖ Distance par rapport à l'espace de visage

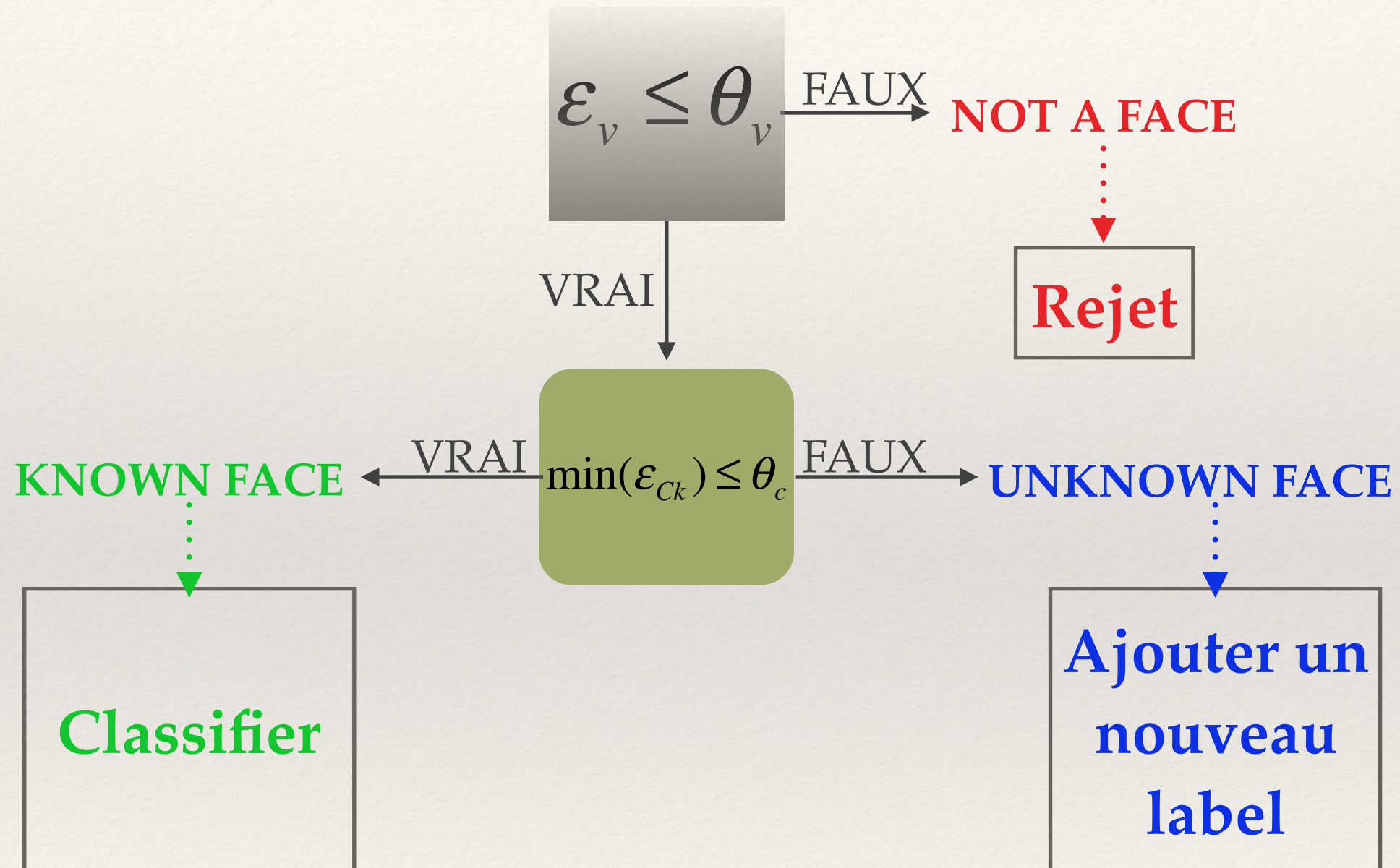
$$\varepsilon_v = d(\Gamma_i, E_v) = \left\| \Gamma_i - \Gamma_i^{proj} \right\|$$

Classification

Projection moyenne de chaque classe :



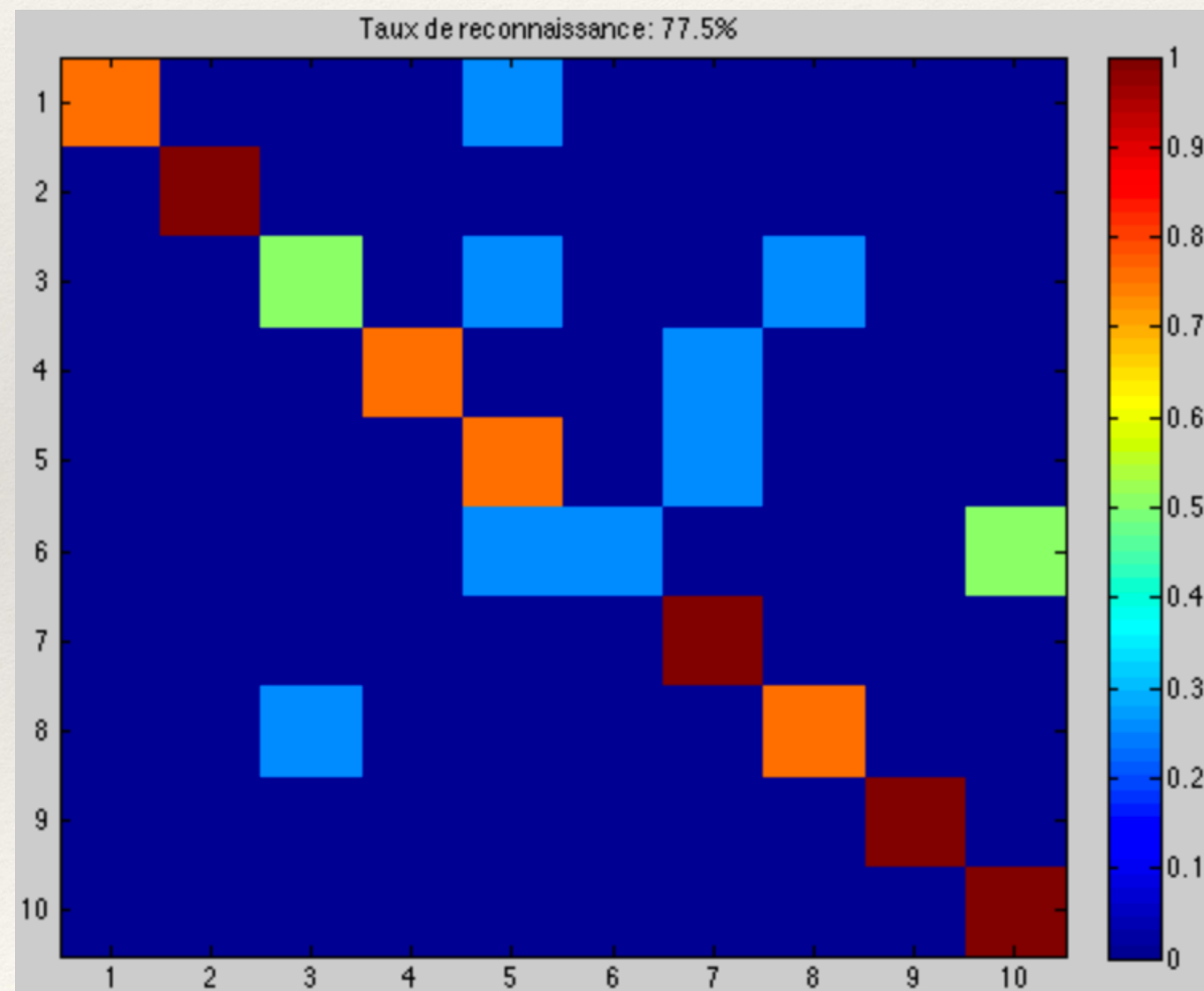
Classification



Classification

Matrice de confusion pour :

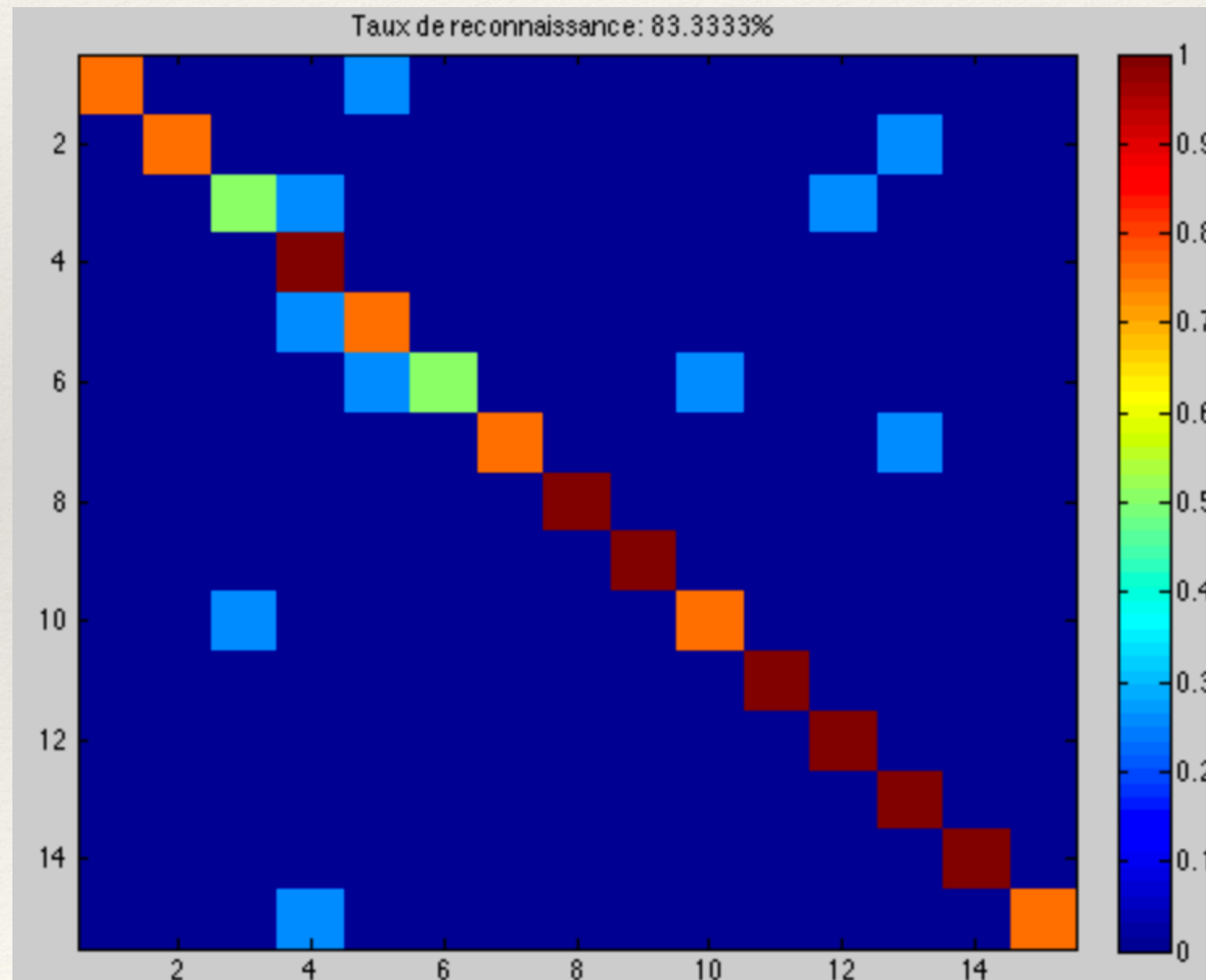
- 10 classes
- $k = 35$ Eigen-faces



Classification

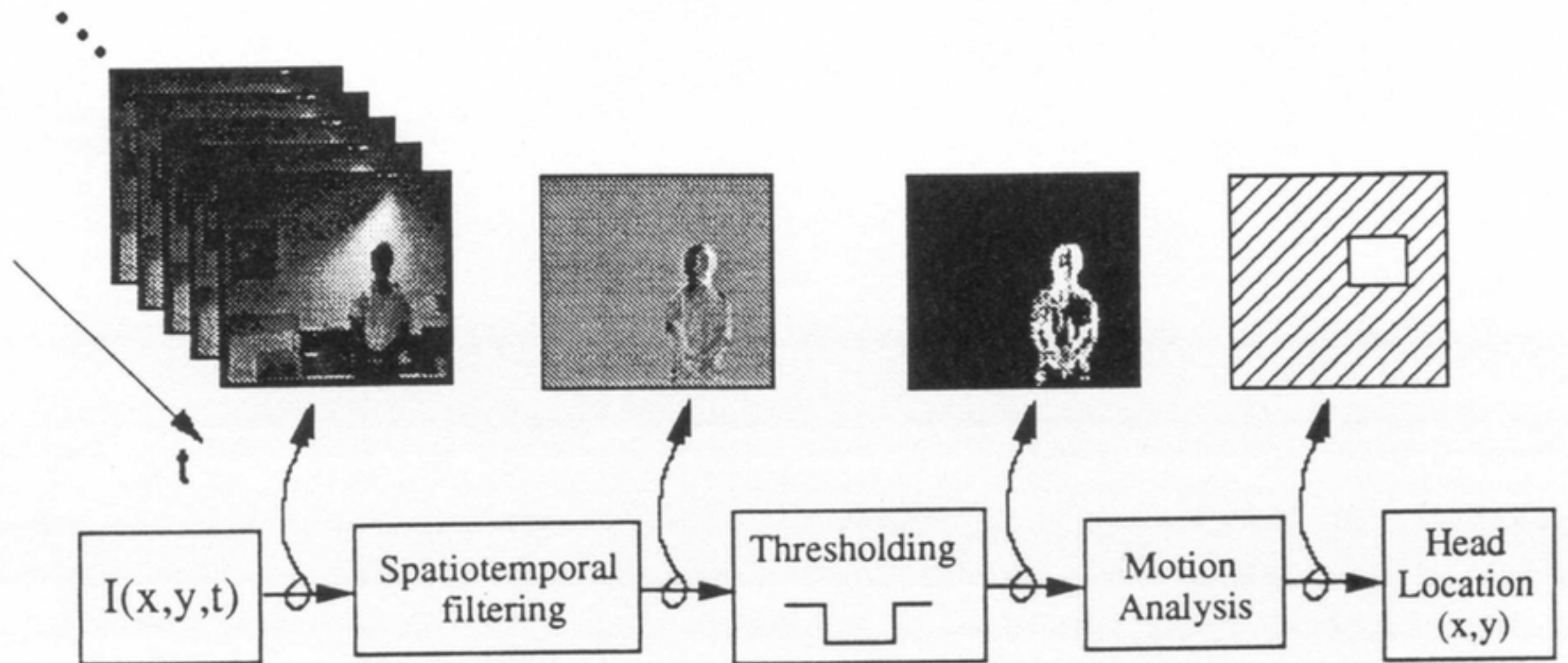
Matrice de confusion pour :

- 15 classes
- $k = 48$ Eigen-faces



Localisation et détection

Détection de mouvement



Face-map

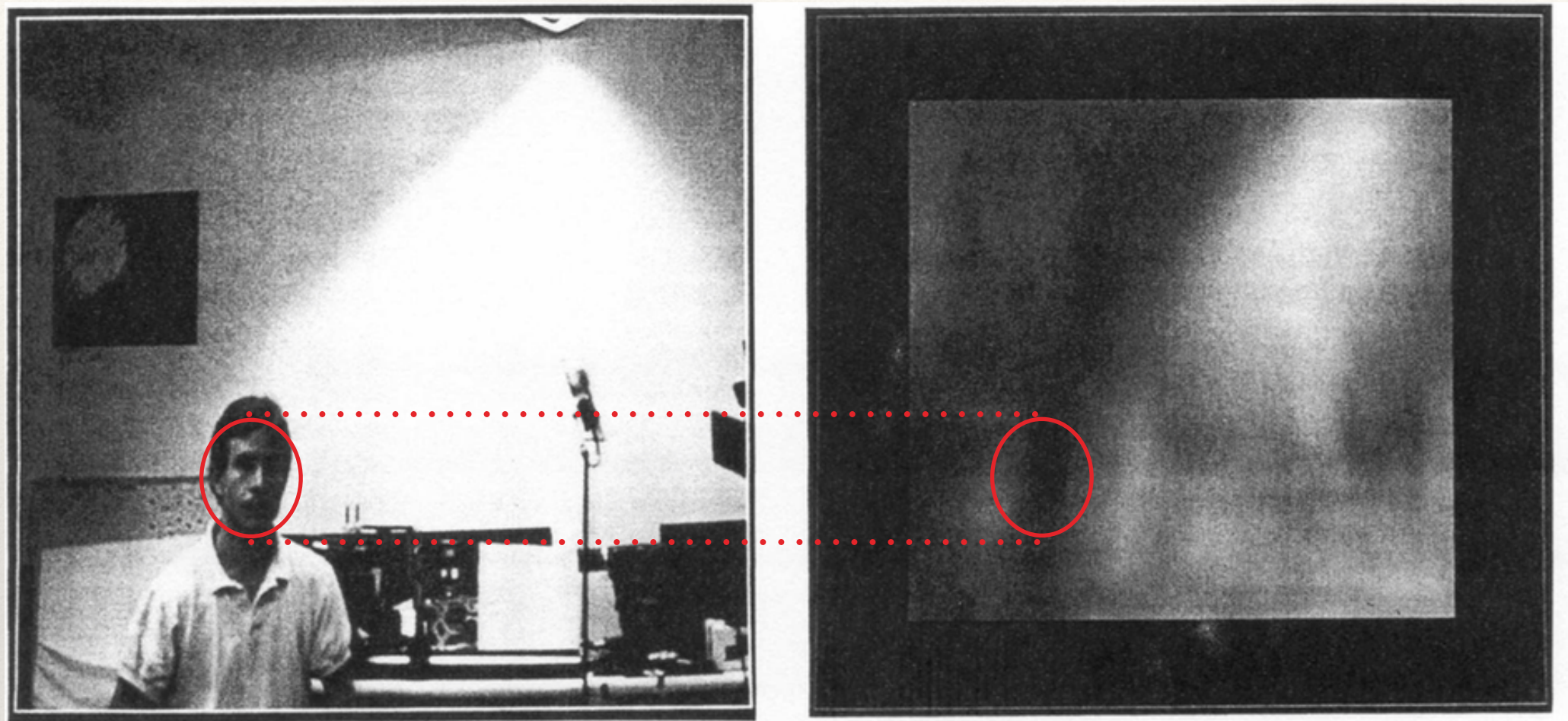
- ❖ Utilisation de l'espace de visage pour localiser
- ❖ La distance par rapport à l'espace de visage s'écrit :

$$\varepsilon_v = d(\Gamma_i, E_v) = \|\Gamma_i - \Gamma_i^{proj}\|$$

En développant un peu l'expression on peut généraliser cette formule pour calculer pour chaque pixel d'une image la distance par rapport à cet espace.

Face-map : $\varepsilon^2(x, y) = \Gamma^T(x, y)\Gamma(x, y) - 2\Gamma(x, y) \otimes \Psi + \Psi^T \Psi + \sum_{i=1}^k [\Gamma(x, y) \otimes u_i - \Psi \otimes u_i]$

Face-map



Invariance en échelle : Resize la sous-image ayant la plus faible distance

Invariance en rotation : Détection orientation du visage

Multiple vues : Classe avec différentes vues (frontal, droite, gauche...)

Questions ?