01/06/2015

Jihade TIKA - Morgan LIENARDY Projet Traitements d’images

IMR2

**EIGENFACES**

Reconnaissance faciale

# Introduction

Le projet consiste à proposer un *système de reconnaissance automatique de visage(s)*, basée sur l’approche *eigenfaces*.

Le principe de la reconnaissance faciale est très simple : comme son nom l’indique, il s’agit d’identifier une face donnée. Selon les algorithmes et les buts recherchés, cette identification peut prendre plusieurs aspects: il peut s’agir de déterminer à qui appartient un visage, de décider si oui ou non ce visage est reconnu, ou même dans certains cas de déterminer s’il s’agit bien d’un visage.

On cherche à spécifier le travail à réaliser en représentant sous forme de blocs les différentes fonctionnalités à implémenter pour cette reconnaissance de visages.

Dans un premier temps on parlera de l’étude préliminaire quant à *eigenfaces*, puis dans un second temps la schématisation des fonctions de traitements.

# Étude préliminaire

L’approche « *eigenfaces »*

Eigenfaces

Initialisation

BDD

Classification --> Résultat

Insertion d’image

Initialisation

Choix des méthodes

# Spécification

Nous spécifions le système en termes de blocs fonctionnels, à différents niveaux de granularité, afin que chaque solution proposée soit validée antérieurement à toute phase de codage proprement dite.

1) Traitement des données en base

2) Ajout d’un visage

Résultat

3) Classification

Base de données

Moyenne

Covariance

SVD

Eigenface

Weight

Distance euclidienne

**Comparaison par rapport à 2 seuils**

Descripteur

1. Traitement des données en base

Base de données

Moyenne

Covariance

SVD

Descripteur

Nous considérons **M** : les images en base, et **p** : les pixels d’une image.

Base de données

Matrice de base B = [p, M]

Matrice avec M colonnes et p lignes. Correspond aux images et leurs pixels

Moyenne

Vecteur moyenne m = [p, 1]

Plus qu’une colonne avec la moyenne de chaque pixel

Covariance

Matrice de covariance C = [p, p]

Avec C = m.m’ (covariance sous scilab)

SVD

3 Matrices de sortie (U, V, S) et U eigenvector = [M, p]

Ne pas oublier de prendre que les 48 premières images

Descripteur

Projection des images sur les eigenfaces d = [M, 48]

Avec d = (B – m).U

2) Ajout d’un visage

Eigenface

Weight

Distance euclidienne

Nous considérons **v** : un nouveau visage

Eigenface

Uinput eigenvector = [1, p]

On récupère les composantes eigenface du nouveau visage

Weight

W = [1, p]]

Compare our input image with our mean image and multiply their difference with each eigenvector

🡪 W = U.(v – m)

Distance euclidienne

Ƹk = | mk - m |²