Le traitement d'images



Reconnaissance de visages par la méthode des eigenfaces





La méthode des eigenfaces

Opencv 2.4 offre une nouvelle classe de reconnaissance faciale appelée FaceRecognizer.

Les algorithmes de reconnaissance possibles pour cette nouvelle classe sont:

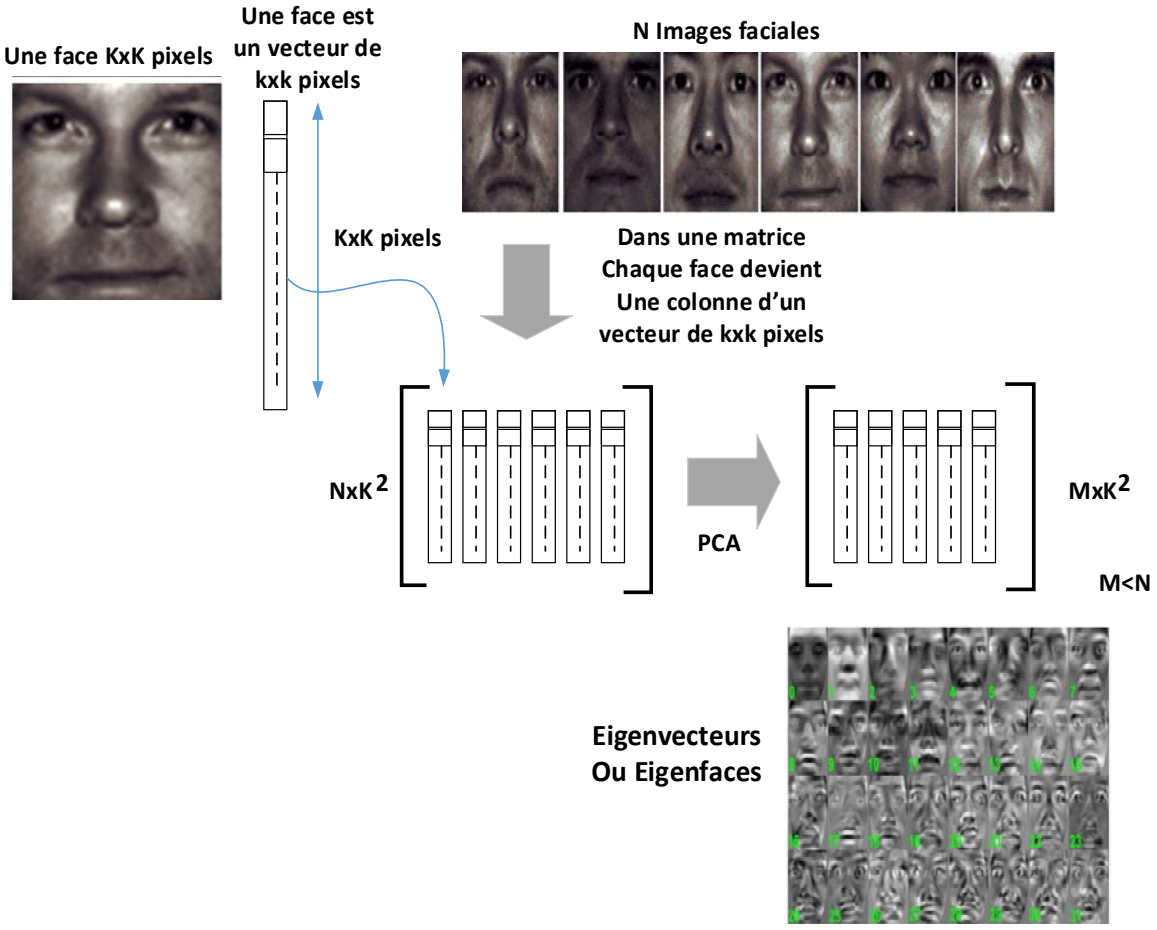
- Eigenfaces (createEigenFaceRecognizer())
- Fisherfaces (createFisherFaceRecognizer())
- Local Binary Patterns Histograms (createLBPHFaceRecognizer())

Afin de proposer une méthode rapide de reconnaissance, un PCA (Principal Component Analysis) est utilisé. Ce PCA applique des méthodes mathématiques afin de transformer un certaines nombre de variables corrélées dans un plus petit nombre de variables décolérées appelées principal components. On dit qu'un PCA va trouver un sous espace de plus petite dimension.

L'utilisation du PCA Principal Component Analysis permet de convertir les images de la liste l'apprentissage en une somme d'"Eigenfaces" qui représente les principales différences entre les personnes ou les images de la liste d'apprentissage.

1) Le principe général

Le PCA va permettre de transformer un espace vectoriel de N images dans un sous espace de dimension inférieure :



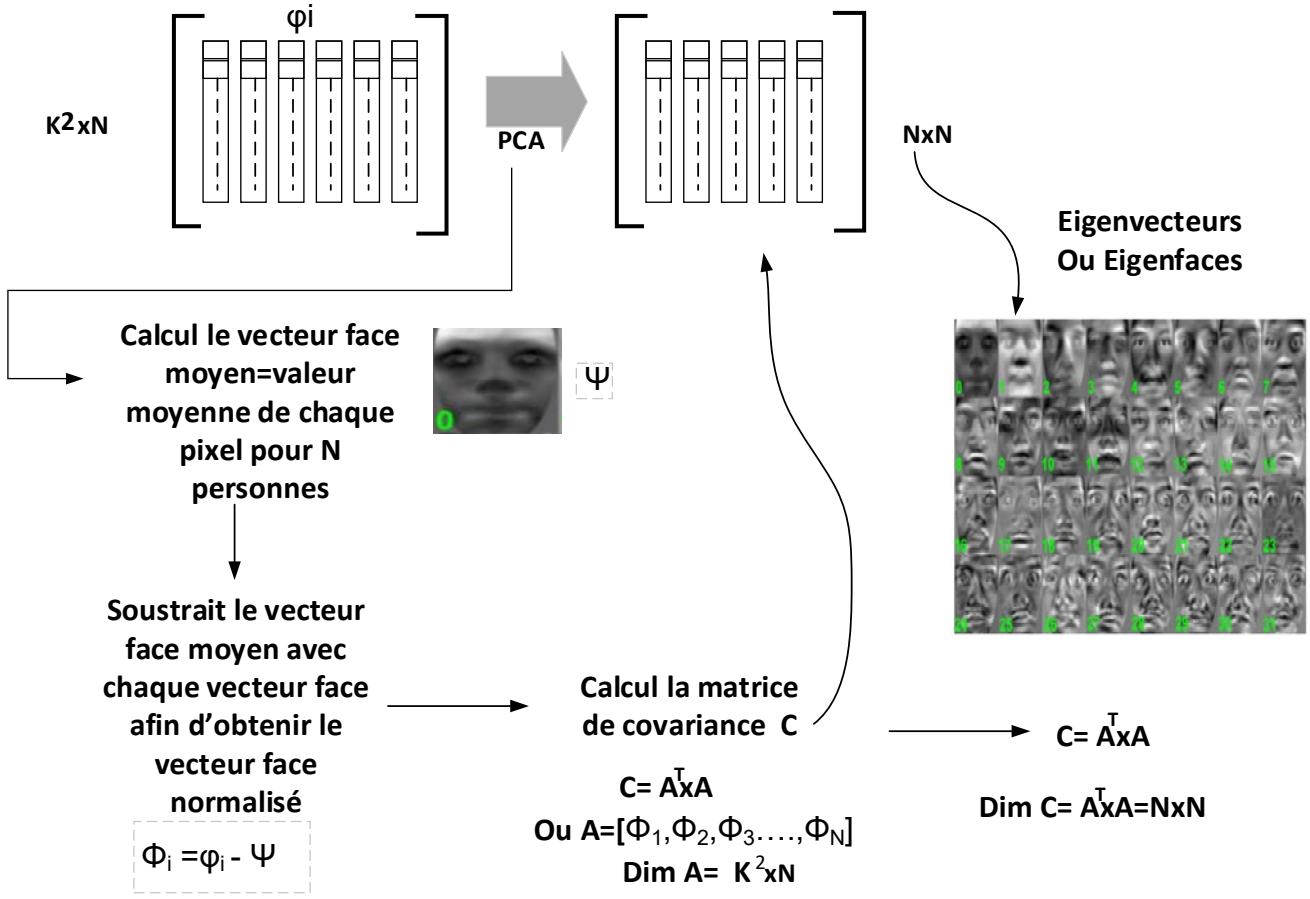
2) Le travail du PCA

Dans un premier temps le PCA trouvera la valeur moyenne de chaque pixel pour les N personnes.

Les eigenfaces sont calculés en comparaison à cette face moyenne (normalisation des images), ou le premier eigenfaces est la différence de face la plus dominante, le second eigenfaces est la seconde différence de faces la plus dominante et ainsi de suite jusqu'à un nombre défini d'eigenfaces qui représente le plus de différence dans une liste d'images d'apprentissage

Le travail du PCA de façon simple est de trouver sous formes d'Eigenfaces les principales différences entre toutes les images de la liste d'apprentissage.

Le diagramme ci-dessous nous expose de façon plus complète les méthodes de calcul du PCA

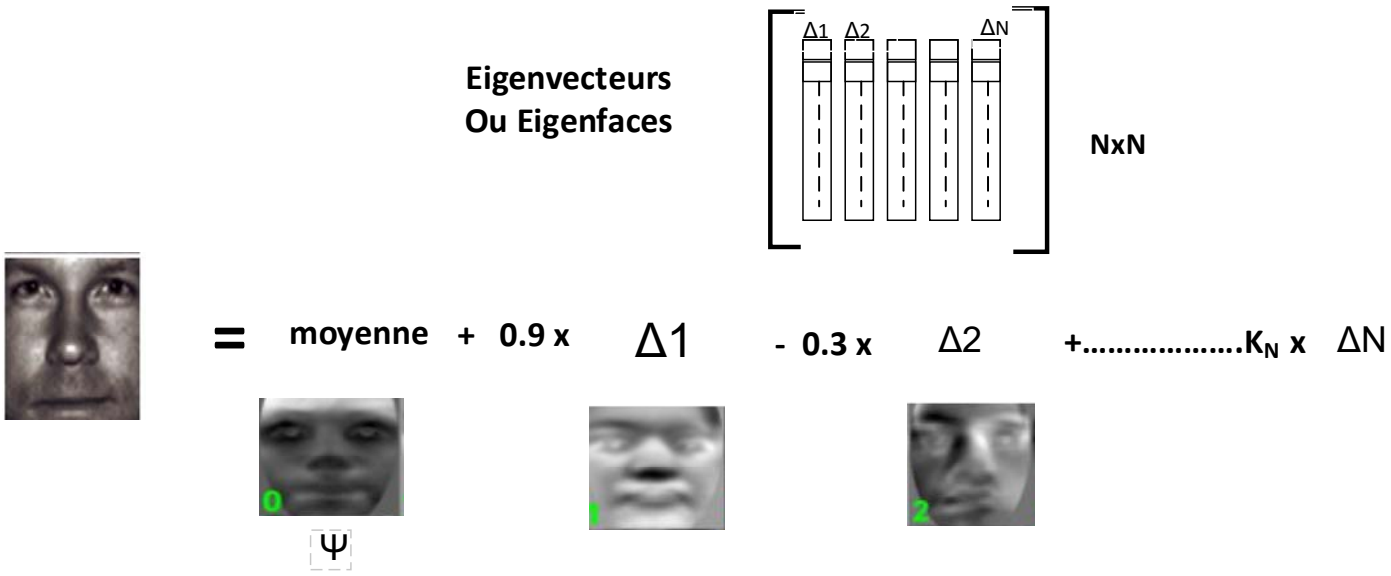


3) La combinaison linéaire

On représente chaque image de la liste d'apprentissage comme la combinaison de ces différences:
Par exemple une des images de la série peut être constituée des ratios suivants:

averageFace + (13.5% de eigenface0) - (34.3% de eigenface1) + (4.7% de eigenface2) + ... + (0.0% de eigenfacen).
Une fois ces combinaisons trouvées, le résultat est le suivant pour une image:
{13.5, -34.3, 4.7, ..., 0.0}.

Le diagramme suivant nous schématise la combinaison linéaire :



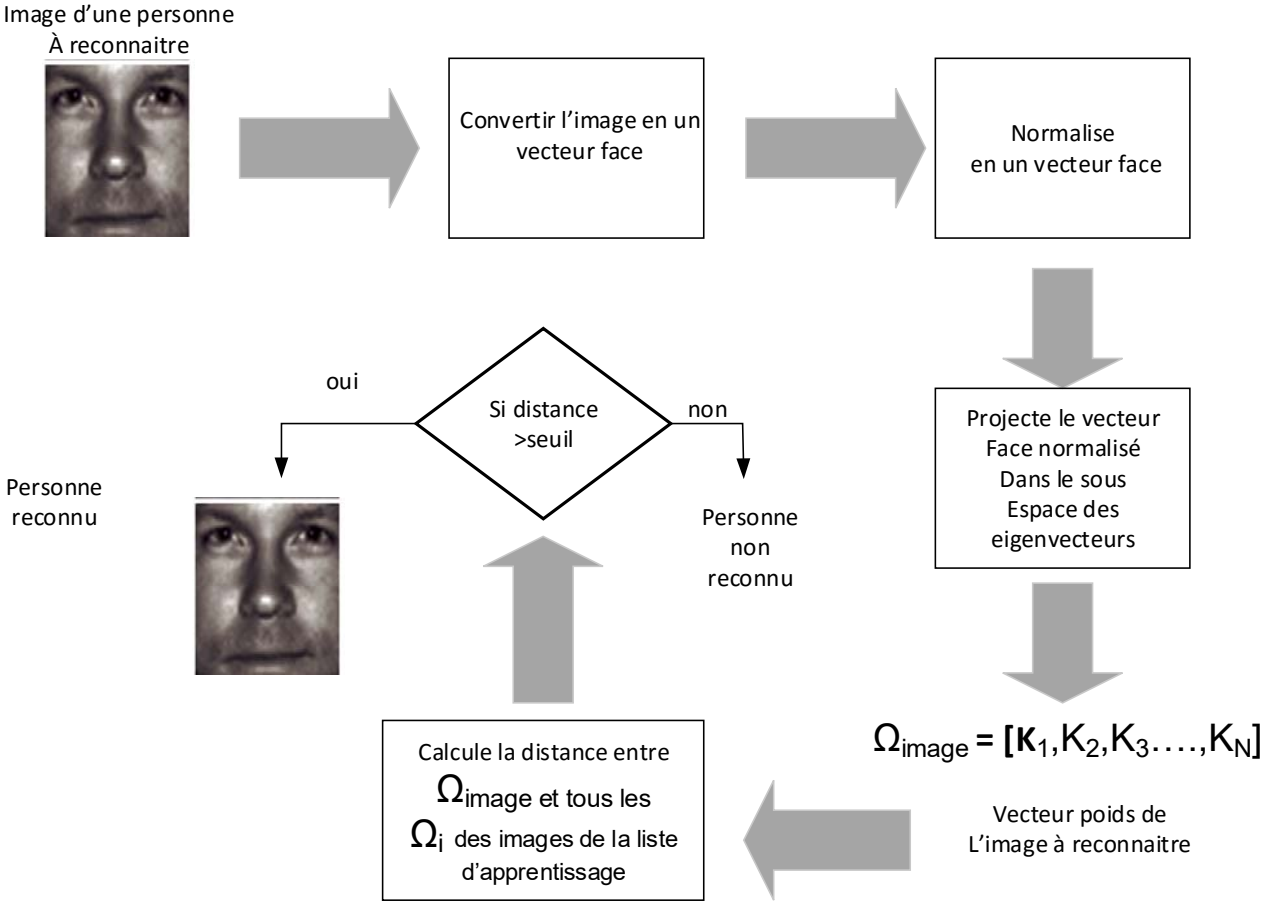
L'image en cours de reconnaissance est une combinaison linéaires des eigenvecteurs dans ce sous-espace

Ou $\Omega_i = [K_1, K_2, K_3 \ldots, K_N]$

Le vecteur poids Ωi est calculé pour chacune des images de la liste d'apprentissage (N images) et pour l'image en cours de reconnaissance par la suite

4) La méthode générale de reconnaissance

Pour reconnaitre une personne, le PCA utilise le même calcul afin d'obtenir une combinaison de ratio en utilisant les mêmes eigenfaces. Cette combinaison est comparée à celle obtenu pour les personnes de la liste d'apprentissage.
La combinaison de ratio la plus proche entre la personne en phase de reconnaissance et les personnes de la liste d'apprentissage nous donne le résultat de cette comparaison.



Basiquement pour créer une base de données pour un lot de personnes et d'images d'apprentissage, on doit créer un fichier texte de métadonnées (faces.csv) qui contient le chemin du fichier image de toutes les personnes en phase d'apprentissage:

1 Shervin data\Shervin\Shervin1.bmp
1 Shervin data\Shervin\Shervin2.bmp
1 Shervin data\Shervin\Shervin3.bmp
1 Shervin data\Shervin\Shervin4.bmp
2 Chandan data\Chandan\Chandan1.bmp
2 Chandan data\Chandan\Chandan2.bmp
2 Chandan data\Chandan\Chandan3.bmp
2 Chandan data\Chandan\Chandan4.bmp