Utilisation des Descripteurs SIFT pour la Reconnaissance Faciale

M'Hand Kedjar
Département de Génie Logiciel
et des Technologies de l'Information
École de Technologie Supérieure
Montréal, Canada
Email: mhand.kedjar.1@ens.etsmtl.ca

***Résumé*—L'identification de personnes par la reconnaissance automatique du visage, de par son immense intérêt pratique et son caractère non intrusif, est l'un des domaines les plus étudiés par la communauté scientifique. Une multitude d'algorithmes ont été développés avec différentes approches pour s'affranchir des problèmes de détection, entre autres le facteur d'échelle, l'occlusion et les variations de pose et d'illumination. Le descripteur SIFT (Scale Invariant Feature Transform), qui a été appliqué avec succès pour la reconnaissance d'objets et la fusion d'images, est utilisé dans cet article pour l'identification faciale. En premier, la région principale d'un visage est détectée à partir d'images en utilisant la méthode mise au point par Viola et Jones. Puis, les descripteurs du visage sont extraits en utilisant SIFT qui consiste en la détection d'extrema, la localisation de points d'intérêt et l'assignation de l'orientation. Ensuite, dans l'étape de mise en correspondance, la reconnaissance du visage est menée en comparant les caractéristiques réelles extraites avec l'ensemble d'apprentissage en utilisant l'algorithme du plus proche voisin. La méthode proposée est comparée à deux autres algorithmes, l'un est basé sur l'analyse en composantes principales (PCA) et l'autre utilise les descripteurs HOG (Histogram of Oriented Gradients) avec un classifieur SVM (Support Vector Machines). Les résultats expérimentaux très prometteurs lors des essais sur les bases de données ORL, YALE et Caltech montrent que la méthode SIFT est assez robuste dans la description du visage. Ainsi, le modèle développé peut facilement être intégré dans divers systèmes automatisés à des fins d'identification ou d'authentification des personnes.**

I. Introduction

Le but de la reconnaissance faciale(RF) est d'identifier ou vérifier un ou plusieurs individus à partir d'images fixes ou des séquences animées en utilisant une base de données de visages. Les techniques biométriques utilisées couramment présentent plusieurs inconvénients. La reconnaissance par l'iris est très précise, mais son implémentation est très dispendieuse sur une large échelle et généralement n'est pas acceptée par les individus. L'identification par les empreintes digitales est une technique biométrique fiable et non intrusive, mais ne convient pas pour les personnes non collaboratives. La reconnaissance du visage semble être un bon compromis entre la fiabilité et l'acceptation par les individus.

La reconnaissance faciale a attiré beaucoup d'attention de la part des scientifiques travaillant dans les domaines de la reconnaissance de formes, l'apprentissage machine et la vision par ordinateur grâce à ses nombreux champs d'applications ainsi qu'aux énormes défis théoriques qu'elle représente. Différents algorithmes ont été proposés dans les dernières années parmi lesquels, nous pouvons citer ceux basés sur les approches holistiques et ceux basés sur l'extraction de points caractéristiques. Les premiers comprennent, entre autres, l'Analyse en Composantes Principales (PCA ou Principal Components Analysis)[13] et l'Analyse Discriminante Linéaire (LDA ou Linear Discriminative Analysis)[14], et les seconds sont basés sur le calcul de l'Histogramme de Gradients Orientés (HOG ou Histogram of Oriented Gradients) ou encore sur la transformation de caractéristiques visuelles invariante à l'échelle (SIFT ou Scale Invariant Feature Transform).

PCA est une méthode de projection linéaire qui consiste à effectuer une réduction de dimensionnalité en codant les visages dans une nouvelle base obtenue à partir des vecteurs propres et en conservant les composantes les plus pertinentes. NDA tente d'obtenir un autre sous-espace pour maximiser le quotient de la variance inter-classe par la variance intra-classe. HOG sont des descripteurs de l'image invariants à la rotation 2D et qui ont été utilisés dans divers problèmes en vision par ordinateur, notamment la détection des piétons[15]. Ils ont également été appliqués avec succès au problème de reconnaissance faciale[16]. Les Machines à vecteurs de support (SVM ou Support Vector Machines)[17] sont une classe d'algorithmes d'apprentissage supervisé qui peuvent être utilisés pour des problèmes de classifications ou de régression. Leur but principal est de trouver un hyperplan qui va séparer les données et maximiser la distance entre les classes. Plusieurs systèmes de reconnaissance faciale[18][19] ont incorporé les SVM dans l'étape d'apprentissage. La méthode basée sur l'extraction de vecteurs caractéristiques en utilisant SIFT a été appliquée avec succès dans divers systèmes de reconnaissance d'objets[1][3]. Cette méthode analyse l'image avec une approche pyramidale pour en extraire des vecteurs caractéristiques. Ces vecteurs caractéristiques sont idéalement invariants à la rotation, aux changements d'échelle ainsi qu'à d'autres transformations dans l'image. Plusieurs systèmes de reconnaissance faciale [21][22][23] ont utilisé SIFT pour l'extraction des descripteurs. Les résultats obtenus par leurs auteurs sont très encourageants et souvent largement supérieurs par rapport aux méthodes basées sur PCA, NDA ou HOG.

Cet article est organisé comme suit. Dans la section II, nous allons expliquer le contexte dans lequel s'inscrit notre étude. La section III va exposer les principaux objectifs que nous nous sommes fixés pour mener à bien notre projet. La section IV explique en détail les données et les outils utilisés. La section V est une analyse détaillée des techniques de forage de données qui ont été utilisées. Nous détaillerons les étapes de détection, d'extraction des caractéristiques et de classification. En particulier, l'algorithme de Viola Jones sera expliqué et la méthode SIFT y sera décrite en détail. Les principaux résultats obtenus sont présentés en section VI. Dans la section VII, nous présentons une discussion en comparant nos travaux à quelques articles pertinents. Nous terminerons par la conclusion dans la section VIII.

II. Contexte

Nous avons décidé de tester la robustesse de l'algorithme de reconnaissance faciale développé en utilisant les bases de données ORL (AT&T)[5], YALE[6] et Caltech[4]. Nous avons choisi ces trois bases de données parce que d'une part, elles sont parmi les plus utilisées par la communauté scientifique, et d'autre part, chaque individu est représenté par plusieurs images avec différentes expressions faciales et conditions d'illumination. De plus, elles sont disponibles gratuitement sur Internet et toutes les images ont été soigneusement annotées. Les modèles utilisés dans ce projet pour la détection du visage et l'extraction des caractéristiques, dont l'implémentation est librement disponible sur le Web, sont pour la plupart ceux suggérés par les auteurs originaux (Lowe[1] et Viola et Jones[2]). Plusieurs chercheurs se sont inspirés pour développer de nouveaux systèmes.

III. Objectifs

L'objectif principal de ce projet est de concevoir un système de reconnaissance faciale en utilisant les descripteurs SIFT. Pour y parvenir, nous avons besoin de trois domaines clés: la détection des visages, l'extraction des vecteurs caractéristiques et la classification.

Étant donnée une image, le but de la phase de détection faciale est de déterminer s'il y a des visages dans l'image, et dans l'affirmative retourner la localisation et l'étendue de chaque visage. Les défis associés à l'étape de détection faciale peuvent être attribués aux facteurs suivants:

- **L'occlusion**. Les visages peuvent être partiellement obstrués par d'autres objets. Dans une image avec un groupe de personnes, certains visages peuvent partiellement occulter d'autres visages.
- **L'expression du visage**. L'apparition d'un visage est directement affectée par l'expression faciale de la personne.
- **La pose**. Les photos d'un visage varient en fonction de la position relative entre la caméra et la personne.
- **L'illumination**. Lors de la formation de l'image, des facteurs tels que les conditions d'éclairage et les caractéristiques de l'appareil photo affectent aussi l'apparence du visage.

Pour réaliser cette tâche, nous avons utilisé la technique développée par Viola et Jones [2] qui sera détaillée dans la section V.

Une fois qu'un visage a été détecté, nous aborderons l'étape de l'extraction des vecteurs caractéristiques (ou signatures) du visage. L'objectif de cette phase est d'extraire des informations compactes et pertinentes pour distinguer entre les images de différentes personnes et stables en termes de variations photométriques et géométriques dans les images. Un ou plusieurs vecteurs caractéristiques sont extraits de la région faciale. SIFT est une technique assez robuste pour la tâche générale de la reconnaissance et la détection d'objets, nous avons décidé de l'utiliser pour le système de reconnaissance faciale.

L'étape de classification se base sur une mesure de distance pour pouvoir associer au visage extrait la classe la plus appropriée qui est celle où le taux de similarité est le plus grand par rapport aux autres classes. Un seuil d'acceptation a été défini et la distance Euclidienne est utilisée avec l'approche du plus proche voisin.

Le dernier objectif est de comparer les performances de l'algorithme basé sur SIFT avec deux autres approches. La première est basée sur PCA, et l'autre sur l'utilisation des descripteurs HOG combinés avec un classifieur de type SVM. Nous allons analyser l'influence de la taille de l'ensemble d'apprentissage sur le taux de reconnaissance pour 3 bases de données.

IV. Matériel

Dans cette section nous allons décrire les caractéristiques des données utilisées et les outils logiciels et matériels qui nous ont aidés à implémenter le système de reconnaissance faciale.

A. Données

La base de données ORL (AT&T)[5] contient 10 images différentes pour chacune des 40 personnes, pour un total de 400 images au format PGM. Pour certaines personnes, les images ont été prises à des moments différents, en variant l'éclairage, les expressions et les détails du visage. La base de données YALE [6] est composée de 165 images en niveaux de gris au format GIF de 15 individus. Il y a 11 images par individu, une pour chaque expression faciale. La base de données Caltech [4] contient 450 images couleur au format JPEG de 28 personnes (11 femmes et 17 hommes). Le tableau I résume les principales caractéristiques des bases de données utilisées, avec n_p: le nombre de personnes, n_i: le nombre d'images, n_{min} et n_{max} représentent le nombre minimum et le nombre maximum d'images par personne, respectivement, et enfin le format et la taille des images en pixels.

B. Outils

Pour générer les ensembles d'apprentissage et de test, nous avons utilisé la fonction Matlab $imageSet()$ qui maintient une hiérarchie entre les différentes images. L'étape de la détection du visage a été réalisée avec l'algorithme de Viola et Jones qui présente des performances temps réel en détection tout en

TABLE I
CARACTÉRISTIQUE DES BASES DE DONNÉES UTILISÉES

BD	n_p	n_i	n_{min}	n_{max}	*format* et taille
ORL	40	400	10	10	PGM, 112×92
YALE	15	165	11	11	GIF, 116×98
Caltech	31	450	1	29	JPEG, 896×592

étant extrêmement précis. L'extraction des descripteurs SIFT a été réalisée en utilisant l'algorithme développé par l'auteur de l'article de référence[1], et dont le code est disponible sur son site web [1]. L'algorithme principal est développé en C++, mais une interface Matlab est également fournie. D'autres implémentations open source existent, mais nous avons décidé d'opter pour l'algorithme original, car il inclut un exécutable pour Linux et Windows. Pour l'environnement informatique, nous avons utilisé le langage de programmation Matlab avec les bibliothèques de traitement d'image et de vision par ordinateur. Le système d'exploitation est Linux Ubuntu 14.04 LTS, installé sur une machine virtuelle à partir de Windows 10, avec un processeur Intel Core i5-4770 @ 3.20 GHZ et 16 Go de RAM.

V. MÉTHODES

Nous traitons le problème de la reconnaissance faciale comme un système d'apprentissage supervisé, car les images des bases de données utilisées sont déjà étiquetées. Notre implémentation du modèle de détection est basée sur l'algorithme de Viola et Jones. La phase d'extraction des vecteurs caractéristiques est réalisée avec la méthode SIFT. L'algorithme du plus proche voisin est utilisé pour la phase de classification. Le schéma général du système est illustré en figure 1

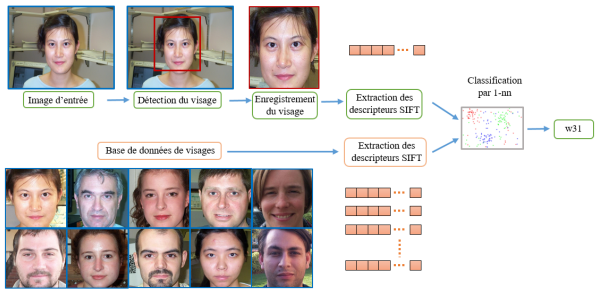

Fig. 1. Schéma du système de reconnaissance faciale

A. Pré-traitement

Les bases de données utilisées étant déjà annotées, nous avons utilisé les fonctionnalités intégrées à Matlab pour vérifier l'intégrité des données et s'assurer que toutes les images sont lisibles et au bon format. Nous nous sommes aussi assurés que les ensembles d'apprentissage et de test contiennent au moins une photo pour chaque individu.

[1]http://www.cs.ubc.ca/~lowe/keypoints/ [accédé le 31-07-2016]

B. Méthodes de forage

Dans cette section, nous allons expliquer les trois principales phases du système de reconnaissance faciale: la détection du visage, l'extraction des descripteurs SIFT et la classification par la méthode du plus proche voisin.

1) Détection faciale: La détection faciale (DF) est le premier étage de tout système automatisé de reconnaissance faciale, puisqu'un visage doit être détecté et localisé avant qu'il puisse être reconnu. Ceci est réalisé grâce à un filtre multi-échelles de Haar en utilisant l'algorithme développé par Paul Viola et Michael Jones en 2001 [2]. Les caractéristiques d'un visage sont décrites dans un fichier xml, et sont construites à partir d'un échantillon de quelques centaines d'images tests. Si un visage est détecté, un rectangle se dessine autour et le résultat est enregistré comme une image.

2) Descripteurs SIFT: La méthode SIFT extrait un ensemble de points d'intérêt (descripteurs) à partir d'une image. Les descripteurs extraits sont idéalement invariants par rapport au changement d'échelle, à la rotation et à la translation. Ils ont aussi la particularité d'être robustes au bruit, flou, contraste, changement du point de prise de vue, déformation de la scène, tout en restant efficaces pour la reconnaissance d'objets [1]. L'algorithme SIFT a été développé par David Lowe en 1999 [8], et une synthèse détaillée est apparue dans l'article de 2004 [1]. Il est composé de 2 étapes principales :

- Calcul des points d'intérêt et extraction des descripteurs.
- Mise en correspondance des points d'intérêt.

a) Description de l'algorithme principal: SIFT détecte une série de points d'intérêt à partir d'une représentation multi-échelle de l'image. Un point d'intérêt est défini par le triplet (x, y, σ), avec (x, y) ses coordonnées sur l'image, et (σ) un facteur d'échelle caractéristique. L'espace des échelles d'une image est défini comme le résultat de convolution d'une image I par un filtre gaussien G de paramètre σ, et il est noté L [9][10]:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{1}$$

où $*$ est l'opérateur de convolution en x et y, et

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} \tag{2}$$

Dans le but d'améliorer la stabilité de la détection des points d'intérêt, Lowe a proposé en 1999 [8] d'intégrer la notion d'espaces d'échelles dans l'évaluation de la différence de Gaussiennes. Il définit ainsi la fonction $D(x, y, \sigma)$ comme la différence de deux espaces d'échelles Gaussiens consécutifs.

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{3}$$

Où k est un facteur multiplicateur constant, généralement pris égal à $\sqrt{2}$. La figure 2 permet de visualiser la construction d'une Différence de Gaussiennes (DoG).

Pour chaque octave dans l'échelle de l'espace, l'image originale est convoluée d'une manière itérative avec une

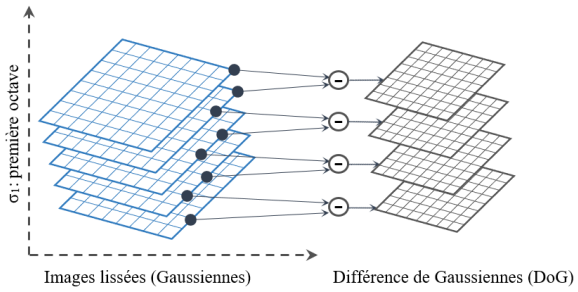

Fig. 2. Différence de Gaussiennes (DoG)

Gaussienne pour produire l'ensemble des échelles de l'espace montrées à gauche de figure 2. Les gaussiennes adjacentes sont soustraites pour produire les différences de Gaussiennes montrées à droite. Après chaque octave, l'image Gaussienne est sous-échantillonnée par un facteur 2, et le processus se répète [1]. Lowe suggère que si nous doublons la taille de la première image en utilisant une interpolation linéaire, avant de bâtir le premier étage de la pyramide des échelles, le nombre de points d'intérêt sera multiplié par 4 [1]. Plus le nombre de points d'intérêt est élevé, meilleur sera la précision de l'algorithme.

b) Détection des extrema dans la différence de Gaussiennes: Une fois que la pyramide des échelles est construite, nous obtenons un lissage de l'image sur plusieurs échelles et à des dimensions différentes. Les images résultantes des différences de Gaussiennes permettent de déterminer les maxima locaux. Un pixel est sélectionné si sa valeur est plus petite (ou plus grande) que tous les 26 autres qui l'entourent (8 dans l'image courante, 9 dans l'image de l'échelle d'en haut, et 9 dans celle d'en bas) [1], comme illustré en figure 3

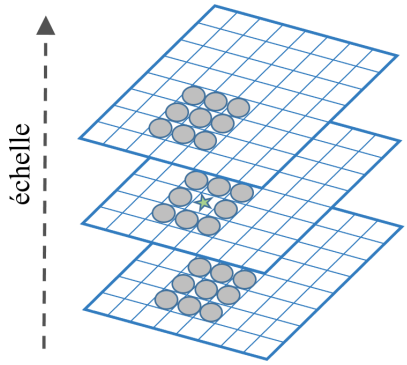

Fig. 3. Détection des maxima

La différence de deux images consécutives lissées par un filtre Gaussien, $D(x, y, \sigma)$, peut être approximée par le filtre LoG (Laplacien of Gaussian), $\sigma^2\nabla^2 G$ comme examiné par Lindeberg (1994) [10]. La relation entre D et $\sigma^2\nabla^2 G$ peut être déduite de l'équation de diffusion de la chaleur:

$$\frac{\partial L}{\partial \sigma} = \sqrt{t}\nabla^2 L \tag{4}$$

En posant: $t = \sigma^2$, l'équation (4) devient:

$$\frac{\partial G}{\partial \sigma} = \sigma\nabla^2 G \tag{5}$$

À partir de là, nous voyons que $\nabla^2 G$ peut être calculé à partir de l'approximation des différences finies de $\frac{\partial G}{\partial \sigma}$ en utilisant les différences des échelles voisines à $k\sigma$ et σ:

$$\sigma\nabla^2 G = \frac{\partial G}{\partial \sigma} \approx \frac{G(x, y, k\sigma) - G(x, y, \sigma)}{k\sigma - \sigma} \tag{6}$$

Et ainsi:

$$G(x, y, k\sigma) - G(x, y, \sigma) \approx (k-1)\sigma^2\nabla^2 G \tag{7}$$

Ceci montre que lorsque la différence de Gaussiennes a des échelles qui diffèrent d'un facteur constant, elle prend déjà en compte le σ^2 qui est requis pour que le Laplacien soit invariant à l'échelle. Le facteur $(k-1)$ dans l'équation est une constante à travers toutes les échelles, et ainsi n'a aucun impact sur la stabilité de la détection des points d'intérêt [1].

c) Localisation précise des points d'intérêt: Une fois qu'un point candidat a été trouvé en le comparant à ses voisins, la prochaine étape est d'effectuer un série de traitements d'images avancés pour le localiser de façon plus précise. Cette étape permet d'éliminer les points qui ont un faible contraste (et qui sont donc plus susceptibles d'être affectés par le bruit), ou qui sont mal localisés autour des bords de l'image. Brown et Lowe (2002) [7] proposent une méthode d'interpolation 3D des coordonnées des points où se trouvent les extrema. Leur approche utilise le développement en séries de Taylor à l'ordre 2 de la fonction $D(x, y, \sigma)$, en prenant comme origine les coordonnées du point candidat:

$$D(x) = D + \frac{\partial D^T}{\partial x}x + \frac{1}{2}x^T\frac{\partial^2 D^T}{\partial x^2}x \tag{8}$$

Où D et ses dérivées sont évaluées au point-clé candidat, et où $x = (x, y, \sigma)^T$ est un offset à partir de ce point. La localisation de l'extremum $\hat{x}$ est déterminée en prenant la dérivée par rapport à x égale à zéro, ce qui donne:

$$\hat{x} = -\frac{\partial^2 D^{-1}}{\partial x^2}\frac{\partial D}{\partial x} \tag{9}$$

La fonction $D(x)$ évaluée au point $D(\hat{x})$ est très utile pour éliminer les points à faible contraste. Ceci est obtenu à partir de:

$$D(\hat{x}) = D + \frac{1}{2}\frac{\partial D^T}{\partial x}\hat{x} \tag{10}$$

Lowe recommande que les extrema ayant une valeur $|D(\hat{x})|$ inférieure à 0.3 soient rejetés (en assumant que les valeurs des pixels sont dans le domaine $[0, 1]$).

Pour des critères de stabilité, rejeter uniquement les points à faible contraste n'est pas suffisant. La fonction différence de Gaussiennes présente une forte réponse autour des contours, ce qui peut donner naissance à des extrema locaux instables même à des niveaux de bruit très bas. La courbure principale

peut être calculée en utilisant la matrice de Hess dans la position et l'échelle du point d'intérêt correspondant:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \tag{11}$$

Où les dérivées $D_{xx}, D_{xy}, \ldots$ sont évaluées en prenant les différences des points d'échantillonnage avec la méthode des différences finies. Les valeurs propres (a, b), avec $a > b$, de la matrice H sont proportionnelles à la courbure principale de D[11]. Le ratio entre a et b, $r = \frac{a}{b}$, est utilisé ($r = 10$ dans l'article de Lowe [1]) comme seuil empirique. Si nous définissons:

$$tr(H) = D_{xx} + D_{yy} \tag{12}$$

$$det(H) = D_{xx} \times D_{yy} - D_{xy}^2 \tag{13}$$

Nous avons:

$$R = \frac{tr(H)}{det(H)} = \frac{(r+1)^2}{r} \tag{14}$$

Tous les points ayant un $R > \frac{(10+1)^2}{10}$ sont éliminés.

d) Assignation de l'orientation: Pour obtenir une invariance par rapport à la rotation, à chaque point d'intérêt retenu après l'étape précédente, sont assignées une ou plusieurs orientations déterminées localement sur l'image à partir de la direction des gradients dans un voisinage autour du point. Le gradient $m(x, y)$ et l'orientation $\theta(x, y)$ sont calculés selon les formules suivantes [1]:

$$m(x,y) = \sqrt{\begin{array}{r}(L(x+1,y) - L(x-1,y))^2 \\ +(L(x,y+1) - L(x,y-1))^2\end{array}} \tag{15}$$

$$\theta(x,y) = \tan^{-1} \frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)} \tag{16}$$

Un histogramme d'orientation de gradient est calculé par rapport au voisinage du point d'intérêt. Il est composé de 36 intervalles couvrant chacun 10 degrés ($36 \times 10^o = 360^o$). Cet histogramme est pondéré, d'une part, par une fenêtre circulaire Gaussienne de paramètre égal à $1,5$ fois le facteur d'échelle du point-clé σ_0[12] et, d'autre part, par l'amplitude de chaque point. Les pics dans l'histogramme obtenu correspondent aux directions dominantes des gradients locaux. Le pic le plus haut et détecté et retenu. De plus, tout pic ayant plus de 80% de la valeur du pic le plus haut est aussi utilisé pour créer un point-clé ayant cette orientation [1]. Ceci est illustré en figure 4.

À l'issue de cette dernière étape, un point-clé est donc défini par quatre paramètres (x, y, σ, θ).

e) Descripteur local type SIFT: Le descripteur est construit à partir de l'image d'échelle la plus proche de l'échelle du point-clé. La fenêtre de la zone d'intérêt est divisée en $n \times n$ blocs. Dans chaque bloc, un histogramme de r valeurs est calculé et les amplitudes des gradients locaux y sont enregistrées. En général, les descripteurs SIFT sont calculés sur une grille de 16×16 points voisins, divisée en 4 blocs

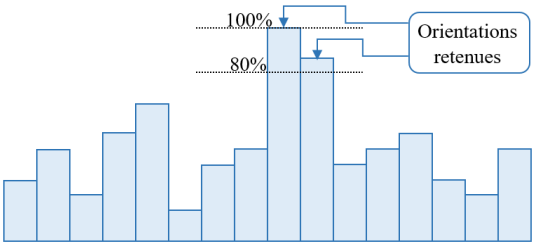

Fig. 4. Histogramme des orientations

de 4×4 et des histogrammes de 8 orientations, et sont donc composés de 128 valeurs [1]. La figure 5 illustre un exemple d'un descripteur SIFT.

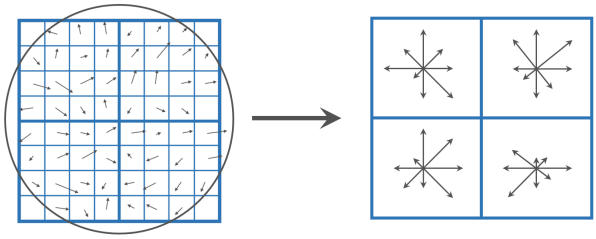

Fig. 5. Exemple d'un descripteur 8×8 et 2×2 (adapté de [1])

3) Classification par l'algorithme du plus proche voisin: Maintenant que nous avons décrit la méthode SIFT et comment les descripteurs sont obtenus, nous allons expliquer comment ils peuvent être utilisés pour la reconnaissance faciale. La méthodologie utilisée pour classifier une nouvelle image est basée sur l'algorithme du plus proche voisin. C'est une des approches les simples et qui consiste à calculer la distance Euclidienne entre toutes les paires de points clés dans les deux images et utiliser la distance minimale pour établir la correspondance. D'un point de vue mathématique, étant donné 2 images I_t et I_r, représentant l'image de test et l'image de référence, respectivement, deux ensembles de points clés sont calculés:

$$K(I_t) = \{k_1^{I_t}, k_2^{I_t}, ..., k_P^{I_t}\}$$

$$K(I_r) = \{k_1^{I_r}, k_2^{I_r}, ..., k_Q^{I_r}\}$$

Nous évaluerons ensuite la distance entre chaque vecteur caractéristique $k_i^{I_t}, i = 1, 2, ..., P$ et tous les vecteurs caractéristiques de $K(I_r)$, et nous prendrons la distance minimale d_i^{min}:

$$d_i^{min} = \min_i d(k_i^{I_t}, k_j^{I_r}), j = 1, 2, ..., Q \tag{17}$$

En posant: $d_1 = d_{min}$, la distance au plus proche voisin, et d_2 la distance au prochain plus proche voisin. Une correspondance entre $k_i^{I_t}$ et $k_j^{I_r}$ est retenue si et seulement si:

$$\frac{d_1}{d_2} < d_s$$

Avec d_s est un seuil de distance préalablement fixé ($d_s = 0.8$ est la valeur optimale retenue par Lowe[1]). Nous obtenons ainsi pour chaque image I_r un certain nombre de points de

correspondance $N_{desc}(I_t, I_r)$. Cette étape est répétée pour toutes les images I_r appartenant à l'ensemble d'apprentissage.

En notant $N_{desc}(I_r^{max}, I_r)$ le plus grand nombre de points de correspondance trouvés, nous aurons alors l'image I_t qui sera associée à la classe de l'image I_r^{max}.

C. Post traitement

Maintenant que les descripteurs SIFT ont été extraits des images, nous allons illustrer quelques applications pour la visualisation des points caractéristiques et la mise en correspondance entre deux images. La figure 6 montre un exemple de l'application de l'algorithme SIFT en utilisant des images des bases de données ORL et Caltech. Pour une image donnée, les descripteurs SIFT sont affichés en vecteurs avec un point d'origine, une amplitude et une orientation.

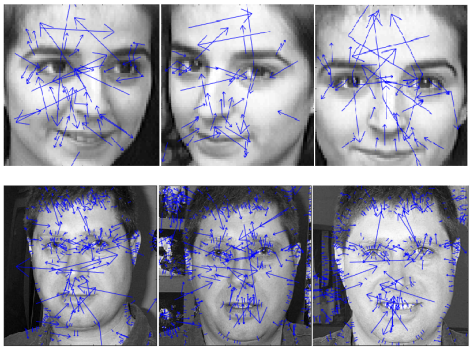

Fig. 6. Exemple d'images avec les descripteurs SIFT extraits

Pour mesurer la robustesse de la méthode SIFT, nous avons pris 2 paires d'images: une appartenant à la même classe, et l'autre appartenant à deux classes différentes. Dans le premier cas, 31 points de correspondance ont été trouvés, contre uniquement 1 dans le second cas. Ceci est illustré en figure 7

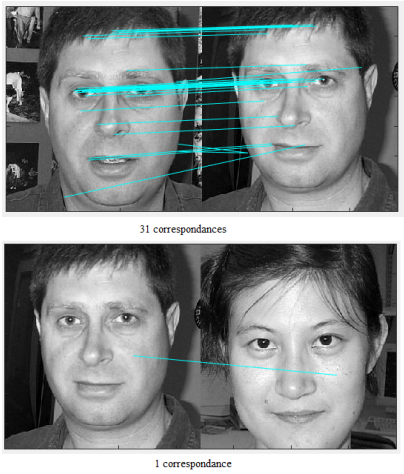

Fig. 7. Exemple de correspondances entre deux images

VI. Résultats

Dans le but de la vérification de la validité et des performances du modèle de reconnaissance faciale proposé, nous avons effectué quelques analyses expérimentales sur les trois bases de données standard ORL, YALE et Caltech. La base ORL est composée de 400 images réparties en 40 classes distinctes. La base YALE contient en tout 165 images de quelque 15 individus. Il y a 28 classes pour la base Caltech totalisant 450 images. Le système développé avec SIFT est comparé à deux autres algorithmes, le premier est basé sur PCA et le second sur l'utilisation des descripteurs HOG avec une classification SVM.

Pour tester la robustesse des trois algorithmes, nous avons développé une méthodologie pour déterminer les ensembles d'apprentissage et de test. Nous définissons le ratio r entre ces deux ensembles par:

$$r = \frac{N_{train}}{N_{total}} = \frac{N_{train}}{N_{train} + N_{test}}$$

Avec N_{train} le nombre d'images dans l'ensemble d'apprentissage, N_{test} le nombre d'images dans l'ensemble de test, et N_{total} le nombre total d'images dans la base de données.

Nous faisons varier r de 0.1 à 0.9, et pour chaque valeur de r, nous répétons l'expérience 10 fois. Il est important de mentionner que dans tous les cas de figure, les ensembles d'apprentissage et de test sont générés aléatoirement. Soit: VP_i le nombre d'images correctement classifiées par l'algorithme à la tentative i, la précision est définie par:

$$prec_i = \frac{VP_i}{N_{total}}, i = 1, 2, ...10$$

La précision finale est calculée comme la moyenne sur les 10 tentatives:

$$prec = \frac{1}{10} \sum_{i=1}^{10} prec_i$$

Le tableau II résume les résultats obtenus au niveau de la précision et de la variance pour chaque algorithme en fonction de la base de données utilisée.

Nous remarquons que la méthode basée sur SIFT donne de meilleurs résultats pour les trois bases de données. La précision moyenne pour ORL est de 98%, ce qui est remarquable. Le résultat SIFT pour Caltech (96%) dépasse de très loin celui avec PCA (31%) et aussi celui obtenu avec HOG+SVM (90%). Pour les trois bases de données, les résultats obtenus avec SIFT sont beaucoup plus stables, comme en témoignent les valeurs de la variance pour les trois cas. Pour la base de données YALE, les résultats sont un peu moins bons (85%), probablement à cause des grands changements d'illumination entre les différentes images.

Pour analyser l'influence de la valeur du ratio r, nous avons tracé le résultat de la précision en fonction de r et du choix de l'algorithme pour la base de données ORL. La figure 8 montre les courbes obtenues.

TABLE II
RÉSULTATS: PRÉCISION ET VARIANCE

BD	ORL		YALE		CALTECH	
	prec	*var*	*prec*	*var*	*prec*	*var*
SIFT	**0.98**	$\mathbf{2.8e^{-4}}$	**0.85**	$\mathbf{19e^{-4}}$	**0.96**	$\mathbf{2.6e^{-4}}$
PCA	0.96	$4e^{-4}$	0.70	$34e^{-4}$	0.31	$27e^{-4}$
HOG+SVM	0.91	$25e^{-4}$	**0.85**	$21e^{-4}$	0.90	$8e^{-4}$

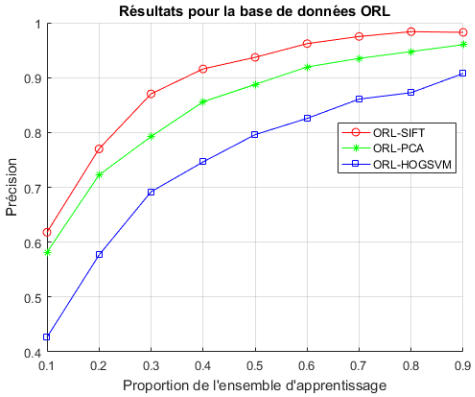

Fig. 8. Résultats pour la base de données ORL

Nous remarquons que la précision des trois algorithmes s'améliore à mesure que r augmente. Nous constatons aussi que la méthode basée sur SIFT donne les meilleurs résultats pour toutes les valeurs de r. Le seuil de 90% de précision est atteint pour une valeur de $r = 0.4$, alors que pour HOG+SVM, il est seulement atteint pour $r = 0.9$. Nous notons également que la méthode basée sur PCA est plus robuste que HOG+SVM.

Dans le but de vérifier la stabilité des trois algorithmes, nous avons tracé les courbes statistiques des 10 différentes mesures pour chaque valeur de r afin d'illustrer la moyenne de chaque mesure ainsi que sa variance. Une grande variance nous informe que l'algorithme n'est pas très stable, et ainsi les valeurs de la précision obtenues peuvent avoir de grandes variations. Nous avons utilisé la fonction $boxplot()$ de Matlab pour obtenir les courbes qui sont tracées en figure 9.

Nous constatons que la méthode basée sur SIFT est la plus stable des trois, et ce pour la plupart des valeurs de r. La méthode PCA est également relativement stable contrairement à HOG+SVM où nous remarquons une grande variabilité des résultats.

Dans cet exemple, nous avons décidé de tester l'algorithme sur la base de données Caltech, qui est une base un peu plus complexe que la base ORL. Une fois de plus, la méthode SIFT donne de loin les meilleurs résultats pour toutes les valeurs de r. Ceci est vrai si nous tenons compte de la précision moyenne, mais aussi de la stabilité au niveau des résultats.

Maintenant que nous avons validé la précision des trois algorithmes, nous avons besoin de mesurer leur temps d'exécution afin de vérifier dans quelle mesure l'algorithme développé peut être intégré dans des systèmes de reconnaissance faciale en temps réel. Le tableau III montre le temps de calcul en secondes obtenu en prenant $r = 0.8$ pour les trois bases de données ORL, YALE et Caltech. Ces valeurs correspondent au temps total de l'exécution de l'algorithme, incluant les phases d'apprentissage et de test, et en utilisant toutes les images de la base de données.

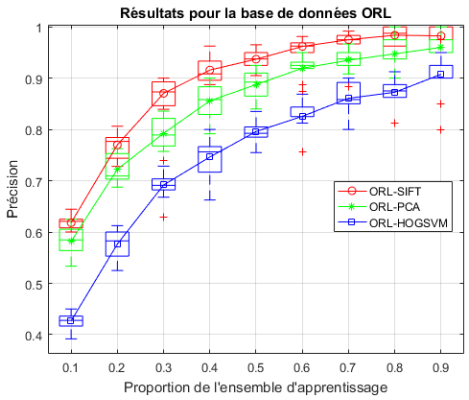

Fig. 9. Boxplot pour la base de données ORL

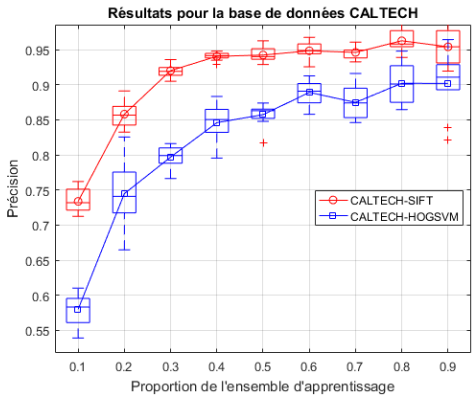

Fig. 10. Boxplot pour la base de données CALTECH

TABLE III
RÉSULTATS: TEMPS DE CALCUL

BD	ORL	YALE	CALTECH
SIFT	441.51	202.02	2329.19
PCA	**1.89**	**0.48**	**48.36**
HOG+SVM	469.43	189.78	519.96

Nous constatons que la méthode PCA est celle qui donne le temps d'exécution le plus bas. SIFT et HOG+SVM présentent des temps de calcul assez élevés, mais il est important de noter que SIFT présente un avantage indéniable: les descripteurs peuvent être calculés une seule fois, stockés dans une base de données et utilisés à chaque fois qu'une nouvelle image doit être classifiée.

Pour quantifier l'influence du paramètre r sur le temps de calcul, nous avons testé les trois algorithmes sur la base de données ORL. La figure 11 montre les résultats obtenus. Nous

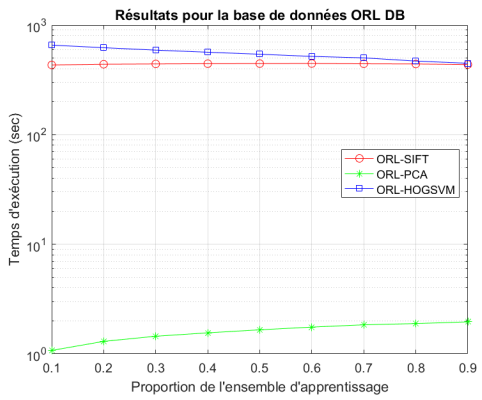

Fig. 11. Temps de calcul pour ORL DB

remarquons que l'algorithme basé sur PCA est de loin le plus performant. Sa vitesse d'exécution est de 2 ordres de grandeur plus rapide que les deux autres algorithmes, SIFT et HOG+SVM.

VII. Discussion

La méthode SIFT est certainement l'algorithme le plus utilisé pour l'extraction de points caractéristiques à partir d'images pour des applications générales de reconnaissance de formes. La raison de cette immense popularité est que les descripteurs SIFT sont invariants à la plupart des transformations présentes dans l'image. Ceci inclut entre autres, la rotation, la translation, le facteur d'échelle et les changements d'illumination. Ils ont aussi démontré leur très grande robustesse face au bruit, flou, contraste, changement du point de prise de vue et à la déformation de la scène. De ce fait, ils sont le choix par excellence dans les domaines de la reconnaissance d'objets, de la fusion d'images, de la stéréo correspondance, pour ne citer que les plus importants. Plus récemment, ils ont regagné une certaine popularité dans le domaine de la biométrie et nous les retrouvons actuellement dans divers systèmes de reconnaissance faciale et plusieurs auteurs ont utilisé SIFT comme base pour l'extraction des points caractéristiques.

Dans [21], les auteurs ont proposé deux nouvelles approches pour appliquer SIFT au problème de reconnaissance faciale: la Keypoints-Preserving-SIFT (KPSIFT) qui garde tous les points clés initiaux comme points caractéristiques, et Partial-Descriptor-SIFT (PDSIFT) où les points clés sur une grande échelle et proches du bord de l'image sont décrits par un descripteur partiel. Avant d'appliquer l'algorithme, les auteurs ont rogné les images pour avoir une taille de 50×57, et ils ont utilisé la similarité cosinus comme mesure de distance. Dans notre étude la taille originale de l'image 92×112 est gardée et la distance Euclidienne est utilisée. De plus, les auteurs de l'article n'ont pas mentionné s'ils ont varié la valeur du ratio entre l'ensemble d'apprentissage et de l'ensemble test. Les performances de leur algorithme sur la base ORL (97%) sont moins bonnes que les résultats que nous avons obtenus dans ce projet (98%).

Dans [22], les auteurs ont utilisé les descripteurs SIFT avec un classifieur type réseau de neurones MLP (multi layer perceptron). La classification d'une nouvelle image est déterminée en comptant le plus grand nombre de points clés classifiés dans une certaine classe. Les résultats obtenus par les auteurs 98% sont similaires aux nôtres pour la base ORL pour $r = 0.8$, mais notre méthode l'emporte pour $r = 0.6$ et $r = 0.7$ (92.25% et 96.75% respectivement reportés par les auteurs, contre 96.19% et 97.50% pour notre méthode).

Dans l'article [23], les auteurs ont utilisé les descripteurs SIFT avec une approche Bag-of-words avec un classifieur SVM à la sortie. Ils ont utilisé une nouvelle mesure de similarité CSID(Cauchy-Schwartz Inequality Distance) pour déterminer à quel cluster assigner un point descripteur. Les auteurs ont testé leur algorithme sur la base ORL avec une précision finale de 93.49% largement inférieure à ce que nous avons obtenu (98%).

Tous les systèmes mentionnés n'ont pas réussi à atteindre un niveau de classification de 100% en utilisant la base de données ORL. Cette base est une des plus simples qui puisse être utilisée, car les visages des individus sont parfaitement alignés, et chaque classe contient plusieurs images. Probablement, de nouveaux algorithmes qui combineront les descripteurs locaux avec les descripteurs globaux (comme la texture, la couleur, et les relations spatiales) vont avoir de meilleurs résultats pour ce qui est de la reconnaissance faciale.

VIII. Conclusion

Dans ce projet, nous avons développé un système de reconnaissance faciale en utilisant les descripteurs SIFT. Nous avons opté pour SIFT car c'est une méthode robuste qui donne d'excellents résultats dans le domaine général de la reconnaissance d'objets tout en étant très utilisée dans l'industrie et au niveau de la recherche scientifique. Le modèle que nous avons développé présente d'excellents résultats pour la reconnaissance des visages dans différentes conditions d'illumination et expressions faciales. Nous avons démontré que même avec un ensemble d'apprentissage assez réduit, nous pouvons atteindre une précision et une stabilité élevées. Nous avons aussi remarqué que les performances de l'algorithme pour la base de données YALE ne sont pas au même niveau que les deux autres bases, ce qui est probablement dû aux grands changements d'illumination entre les différentes images de cette base. La comparaison avec les approches PCA et HOG+SVM a démontré que la méthode SIFT est beaucoup plus appropriée pour les trois bases de données étudiées. Au niveau du temps d'exécution, la méthode PCA est celle qui donne les meilleurs résultats. Pour la méthode SIFT, le temps de calcul est assez correct pour des applications où l'aspect temps réel n'est pas très important. Extraire d'une image des centaines, voire des milliers, de vecteurs de 128 composantes, puis effectuer une comparaison avec une base de données demande énormément de temps de calcul. De nouvelles méthodes basées sur le GPU[20]pour le calcul des descripteurs SIFT sont proposées récemment, et avec la puissance des nouvelles cartes graphiques actuelles, le temps d'exécution

de la méthode SIFT sera considérablement amélioré. Dans l'avenir, nous proposons d'utiliser la méthode développée dans ce projet pour l'identification automatique du sexe et de l'âge d'une personne à partir d'une photo de son visage. C'est un domaine en plein essor, et énormément d'applications pratiques sont possibles au niveau de la personnalisation des écrans de visualisation, comme ceux des téléphones intelligents, des téléviseurs UHD-HDR (Ultra HD, High Dynamic Range), et des écrans intégrés dans les véhicules, en fonction de l'utilisateur.

REMERCIEMENTS

Je tiens à remercier Mme Sylvie Ratté ainsi que Mr Éric Velasquez Godinez pour leur écoute et leur accompagnement durant le cours MTI830 ainsi que leurs conseils judicieux pour le choix et l'accomplissement de ce projet.

REFERENCES

[1] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," International Journal of Computer Vision, vol.60, no.2, pp. 91–110, 2004

[2] Paul Viola, Michael Jones. 2001. "Rapid Object Detection using a Boosted Cascade of Simple Features". Conference On Computer Vision And Pattern Recognition 2001

[3] F. A. Pavel, Z. Wang and D. D. Feng, "Reliable object recognition using SIFT features," Multimedia Signal Processing, 2009. MMSP '09. IEEE International Workshop on, Rio De Janeiro, 2009, pp. 1-6.

[4] Caltech database. http://www.robots.ox.ac.uk/~vgg/data3.html [accédé le 31-07-2016].

[5] ORL database. http://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html [accédé le 31-07-2016].

[6] Yale database. http://vision.ucsd.edu/content/yale-face-database[accédé le 31-07-2016].

[7] Brown, M. and Lowe, D.G., "Invariant features from interest point groups." In British Machine Vision Conference, Cardiff, Wales, pp. 656-665, 2002.

[8] Lowe, D.G., "Object recognition from local scale-invariant features," in Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on , vol.2, no., pp.1150-1157 vol.2, 1999

[9] Koenderink, J.J., "The structure of images. Biological Cybernetics," 50:363-396, 1984

[10] Lindeberg, T.. "Scale-space theory: A basic tool for analysing structures at different scales." Journal of Applied Statistics, 21(2):224-270, 1994.

[11] Harris, C. and Stephens, M., "A combined corner and edge detector." In Fourth Alvey Vision Conference, Manchester, UK, pp. 147-151, 1988.

[12] Article SIFT sur Wikipedia: https://fr.wikipedia.org/wiki/Scale-invariant_feature_transform [accédé le 31-07-2016].

[13] Turk. M., Pentland. A., "Eigenfaces for recognition," Journal of Cognitive Neuro-science 3, pp. 71-86, 1991.

[14] Etemad. K., Chellappa. R., "Discriminant analysis for recognition of human face images," Journal of the Optical Society of America 14, pp. 1724-1733, 1997.

[15] Bertozzi, M., Broggi, A., Rose, M.D., Felisa, M., Rakotomamonjy, A., Suard, F., 2007. A pedestrian detector using histograms of oriented gradients and a support vector machine classifier. In: Proc. Intelligent Transportation Systems Conf., pp. 143– 148.

[16] C. Shu, X. Ding and C. Fang, "Histogram of the oriented gradient for face recognition," in Tsinghua Science and Technology, vol. 16, no. 2, pp. 216-224, April 2011.

[17] C. Cortes and V. N. Vapnik. Support-vector networks. Machine Learning, 20(3), 273-297, 1995.

[18] H. Jia and A. M. Martinez, "Support Vector Machines in face recognition with occlusions," Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on, Miami, FL, 2009, pp. 136-141.

[19] C. Wang, L. Lan, Y. Zhang and M. Gu, "Face Recognition Based on Principle Component Analysis and Support Vector Machine," Intelligent Systems and Applications (ISA), 2011 3rd International Workshop on, Wuhan, 2011, pp. 1-4.

[20] Wu C. SiftGPU manual. (Disponible à: http://cs.unc.edu/~ccwu/siftgpu/#Download) [accédé le 31-07-2016]

[21] Cong Geng and X. Jiang, "SIFT features for face recognition," Computer Science and Information Technology, 2009. ICCSIT 2009. 2nd IEEE International Conference on, Beijing, 2009, pp. 598-602.

[22] T. Liu, S. H. Kim, H. S. Lee and H. H. Kim, "Face recognition base on a new design of classifier with SIFT keypoints," Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on, Shanghai, 2009, pp. 366-370.

[23] D. Liu, D. m. Sun and Z. d. Qiu, "Bag-of-Words Vector Quantization Based Face Identification," Electronic Commerce and Security, 2009. ISECS '09. Second International Symposium on, Nanchang, 2009, pp. 29-33