

Rapport

Projet Régie Vidéo Autonome et Mobile

BODARD Christelle - QIAN Jean - ZOMINY Laurent

Sommaire

1) Présentation du projet	3
2) Android	3
a. Prendre la photo	3
b. Détecter le visage	3
c. Send to the robot	Erreur ! Signet non défini.
3) Reconnaissance faciale	4
a. Redimensionnement des images	4
b. Détection	4
c. Reconnaissance	4
4) Suivi	4
5) Zoom	5
6) Problèmes rencontrés	5

1) Présentation du projet

Projet demandé par une Start'up. Leur but est de mettre en place des robots pour des entreprises qui sont capable de filmer des scènes de façon autonome. Pour cela il nous a été demandé de faire un programme étant capable de reconnaître une personne dans une vidéo et de suivre celle-ci. Nous avons donc découpé notre projet comme ceci :

- Ajout d'une personne à reconnaître
- Détection faciale
- Reconnaissance
- Suivi

Ce sont les tâches principales du projet qui a été développé sous Windows et Python. Il y a eu d'autres sous tâches que nous n'avons pas eues le temps de mettre en place.

2) Android

L'application Android est divisée en trois parties:

1. Prendre la photo
2. Détecter le visage
3. Envoyer au robot

a. Prendre la photo

Pour prendre la photo du visage nous pouvons sélectionner des images déjà présentes sur le smartphone ou prendre de nouvelles photos en utilisant la camera.

b. Détecter le visage

On utilise principalement `android.media.FaceDetector` pour identifier les visages dans une image `Bitmap`. On va détecter la position des yeux dans une étape que va suivre.

c. Envoie au PC

L'utilisateur va nommer l'image et envoyer le nom et l'image sur l'ordinateur. Si l'image n'est pas disponible, il y aura un message d'erreur : "No image found! Please add an image!" Si le nom est donné il y aura le message "Name empty! Please input a name"

3) Reconnaissance faciale

La reconnaissance faciale est un des divers domaines de la biométrie. Il y a actuellement beaucoup d'études dessus. Elle n'est pas précise à 100% à cause des variations de lumière, du changement d'expressions du visage, le changement de maquillage, coupe de cheveux, pilosité faciale... Pour la détection et la reconnaissance, nous avons choisi d'utiliser la librairie d'opencv.

a. Redimensionnement des images

Les images récupérées sont transformée par un algorithme qui, selon la position des yeux, tourne l'image pour mettre le visage droit, tronque l'image pour qu'il n'y ai plus que le visage, et la redimensionne pour avoir comme résultat une image de taille fixe.

b. Détection

Pour la détection on a d'abord pensé à créer un classifieur pour chaque personne à détecter. Pour cela il est nécessaire d'avoir un très grand nombre d'images positives, c'est à dire des images contenant l'élément à reconnaître, et un très grand nombre d'images négatives, des images sans l'élément à reconnaître. Cette méthode aurait été extrêmement longue. Nous avons donc préféré utiliser les classifieurs de haar cascade permettant de détecter des visages et des profils.

c. Reconnaissance

Maintenant que nous avons les visages détectés, openCV offre trois méthodes de reconnaissance faciale dans sa classe FaceRecognizer:

- EigenFace
- FisherFace (plus précis que le EigenFace)
- Local Binary Patterns Histograms (beaucoup moins long que les deux autres méthodes).

Nous avons choisi la méthode de reconnaissance LPBH.

Nous avons essayé d'utiliser une autre librairie pour la reconnaissance faciale libccv pour avoir une meilleure précision, mais nous n'avons pas réussi à la faire fonctionner sous l'environnement Windows.

4) Suivi

Nous avons simulé le suivi à l'aide d'une caméra montée sur un servo-moteur et une carte arduino. Un programme python envoie des informations à la carte arduino pour pouvoir bouger le servo moteur pour que la caméra centre au milieu du plan filmé le visage détecté.

5) Zoom

Nous avons essayé d'effectuer un zoom sur le visage détecté.

6) Problèmes rencontrés

Au cours de notre projet nous avons rencontré plusieurs difficultés.

- La compréhension de notre projet
 Au début il nous a été difficile de définir notre mission.
- La compréhension de python et openCV
 Ce sont pour nous de nouveaux outils que l'on a appris à manipuler.
- Les méthodes à utiliser
 Nous ne savions pas par quel chemin faire avancer notre projet.
- Les tentatives qui n'ont pas abouties
 1. Libccv
 2. Ffmpeg : nous avons essayé de l'utiliser pour envoyer la vidéo sur un autre support.
 3. Lissage : qui aurait été utile pour le zoom
 4. ...