

Dossier de présentation de TPE

Robot d'assistance à localisation par reconnaissance
faciale

Quentin Barrière, Grégoire Charleux, Yannis Lamoussière,
Mathis Pastorelli

2016 – 2017

Sommaire

1	Présentation du projet	3
1.1	L'idée	3
1.2	Historique	3
2	Ce que nous avons fait	3
2.1	Mécanique	3
2.2	Électronique	3
2.3	Programmation	3
2.4	Consruction du robot	4
3	Ce qu'il nous reste à faire	5
3.1	Matériel	5
3.2	Programmation	5
3.3	Construction du robot	5

Introduction

1 Présentation du projet

1.1 L'idée

1.2 Historique

2 Ce que nous avons fait

2.1 Mécanique

Grégoire Charleux

2.2 Électronique

Yannis Lamoussière

2.3 Programmation

En ce qui concerne la programmation, le projet utilisera plusieurs langages : Python, pour le code du robot ainsi que pour le programme version ordinateur, Java, pour l'application Android, ainsi que le HTML, le CSS et le PHP pour le site internet, avec sûrement l'intégration de quelques commandes Bash pour la génération de fichier de reconnaissance faciale, et des requêtes SQL pour accéder à la base de données.

En premier lieu, le programme du robot utilisera le langage interprété Python, accompagné de nombreux modules, faisant partie, pour la plupart, de la bibliothèque standard, bien qu'au moins deux n'en fassent pas partie : OpenCV et Bluez. Le premier sert à la reconnaissance faciale ; c'est un module (une librairie) développé par Intel qui permet, entre autres, de reconnaître des visages dans une image (ou, dans ce cas-là, une vidéo), et ce notamment grâce au Deep Learning. Le second module, quant à lui, sert à la gestion du bluetooth ainsi que d'une connexion (bluetooth), ce qui permet de se connecter à un autre appareil bluetooth, tel qu'un smartphone sous Android, ou un ordinateur disposant d'un adaptateur bluetooth. Le programme est pour l'instant divisé en fichiers que j'ai tous codés, sauf mention contraire :

- arduino.py est un fichier contenant des classes (pour l'instant au nombre de quatre: Arduino, Stepper, Servo et Motor) que j'ai créés afin de rendre le code plus facile à comprendre, grâce à la Programmation Orientée

Objet (POO). La POO permet de masquer le fonction d'une fonction ou plus généralement d'un algorithme, ce qui rend le programme plus clair et compréhensible.

- lcd.py est un fichier contenant une classe (Lcd) permettant d'afficher des messages sur un écran LCD. J'ai ainsi suivi un tutoriel expliquant le fonctionnement d'un écran LCD avec un Raspberry Pi et chaque instruction a un rôle précis et est nécessaire au bon fonctionnement de l'écran. L'on pourrait donc considérer cette partie du code comme étant la seule que je n'aurais pas codée par moi-même, cependant je tiens à ajouter que, d'une part, je comprends le rôle de chaque instruction de ce code, et, d'autre part, sa rédaction n'était pas orientée objet, j'ai donc tout de même adapté ce code pour le faire correspondre à une programmation orientée objet.

- menu.py est un fichier contenant également une classe permettant de gérer un menu sur un écran LCD ainsi que de défiler dans ce dernier.

- menu_handler.py est un fichier contenant

- arduino_communication.py est un fichier contenant

- bluetooth_communication.py est un fichier

- wi-fi_communication.py est un fichier

- robot.py

ANDROID

Enfin, en ce qui concerne le site internet, je ne juge pas intéressant de vous expliquer son fonctionnement en détail (dans ce court dossier en particulier ; je l'expliquerai évidemment dans le dossier final à rendre au jury). En effet, il s'agit pour l'instant presque exclusivement d'un ensemble de fichiers php et css dont la seule fonction (pour l'instant) est d'afficher des pages web, ce qui ne présente pas un grand intérêt quant à la compréhension du fonctionnement du robot.

Pour terminer, voici une partie du code du robot, qui est disponible, comme tout les autres fichiers, à l'adresse suivante : <https://thaspdev.github.io/TPE>.

Mathis Pastorelli

2.4 Construction du robot

Quentin Barrière

3 Ce qu'il nous reste à faire

3.1 Matériel

3.2 Programmation

3.3 Construction du robot