

# Barbichette

(en cours)

Installation interactive pour rire et causer des algorithmes de reconnaissance faciale

## Démarrer l'installation v0

Paramètres webcam : guvcview → changer au moins la fréquence de rafraîchissement sur 50Hz

Lancer qt creator : fichier → projet récent → dlibFaceTrackerTEST.qvs Cliquer sur la grosse flèche verte ou utiliser le raccourci clavier ctrl+r

Choisir la webcam à utiliser → 0 si 1 seule webcam Choisir le périphérique audio à utiliser → regarder dans la liste mais sûrement 6 ou 8

Raccourcis clavier f : fullscreen i : indicateur de performance et valeurs des différents types de sourire. u : fait apparaître la souris et plusieurs options

1. radius : largeur des landmarks
2. smile threshold : seuil de détection d'un sourire
3. minimum head size : défini la taille minimum pour qu'un visage soit capté.
4. detection method : 1 - capte le plus grand visage

2 - capte le visage le plus au centre

1. debug mode : fait apparaître l'image de la webcam en arrière plan

## Automate

Construction d'un automate d'après le projet de Rolf Jethon, voir

[https://lesporteslogiques.net/wiki/openatelier/projet/tete\\_animatronique](https://lesporteslogiques.net/wiki/openatelier/projet/tete_animatronique)

L'automate est commandé par un raspberry pi et actionné par des servomoteurs

## Installation d'un ordi pour la reconnaissance faciale

Installation de l'ensemble de logiciels nécessaires sur un client léger (CL28 : i3-4170T @ 3.2 GHz x 2, 4GB RAM, Debian 11 bullseye)

### Installation de Qt Creator

installation de QTCreator par apt (cf. <https://openframeworks.cc/setup/qtcreator/>)

```
sudo apt install qtcreator #version : Qt Creator 4.14.1 based on Qt 5.15.2
sudo apt install qbs
```

### Installation openframeworks

téléchargement du paquet OF v.0.11.2 <https://openframeworks.cc/download/>

installation en suivant <https://openframeworks.cc/setup/linux-install/>

Ci-dessous, OF est à remplacer par le nom du répertoire choisi pour les fichiers d'openframeworks

installer les bibliothèques dépendances

```
cd OF/scripts/linux/ubuntu
sudo ./install_dependencies.sh
```

compiler openframeworks

```
cd OF/scripts/linux
./compileOF.sh -j2 #2 = nombre de cores
```

ensuite, test d'un exemple avec make

```
cd OF/examples/graphics/polygonExample
make
make run
```

compiler PG (le Project Generator pour Qt Creator)

```
cd OF/scripts/linux
./compilePG.sh
```

J'installe aussi le Project Generator en ligne de commande

```
projectGenerator --help
```

installer QT Creator plugin for openframeworks

```
cd OF/scripts/qtcreator/
./install_template.sh
```

J'essaie d'ouvrir le projet polygonExample dans Qt Creator

- fichier / ouvrir un projet
- compiler / compiler le projet
- compiler / exécuter (CTRL-R)

→ ça fonctionne

## Installation caméra

Premiers test avec une caméra PS3 eye

```
sudo apt install v4l-utils
sudo apt install guvcview
sudo apt install webcamoid
```

(?) La caméra plante très vite avec guvcview ou webcamoid, et n'est plus visible par `lsusb`

**Installation d'addons dans openframeworks** cf.

[https://openframeworks.cc/learning/01\\_basics/how\\_to\\_add\\_addon\\_to\\_project/](https://openframeworks.cc/learning/01_basics/how_to_add_addon_to_project/)

- <https://github.com/kylemcdonald/ofxCv/tags>
- (A INSTALLER) <https://github.com/HalfdanJ/ofxFaceTracker2>

Mais un premier exemple d'ofxCv ne compile pas ...

:( à suivre ...

## Installation des bibliothèques spécifiques

À compléter

## Préparation d'un prototype de code

À compléter

## Journal

Le journal est tenu sur la page [tête animatronique](#)

Article extrait de : <http://www.lesporteslogiques.net/wiki/> - **WIKI Les Portes Logiques**  
Adresse : <http://www.lesporteslogiques.net/wiki/projets/barbichette/start?rev=1688051042>  
Article mis à jour: **2023/06/29 17:04**