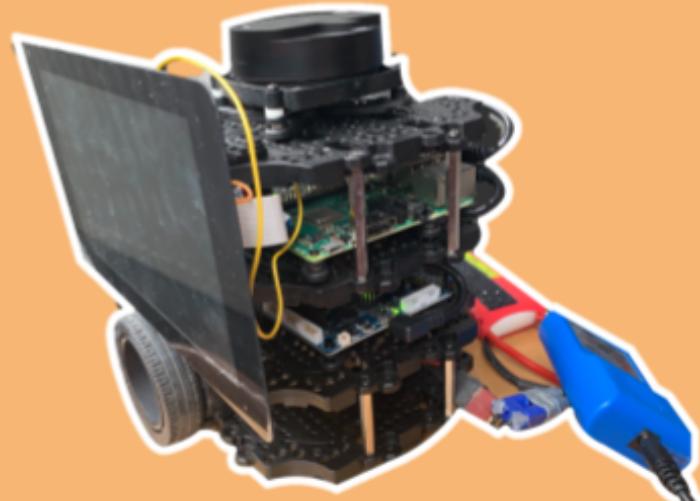
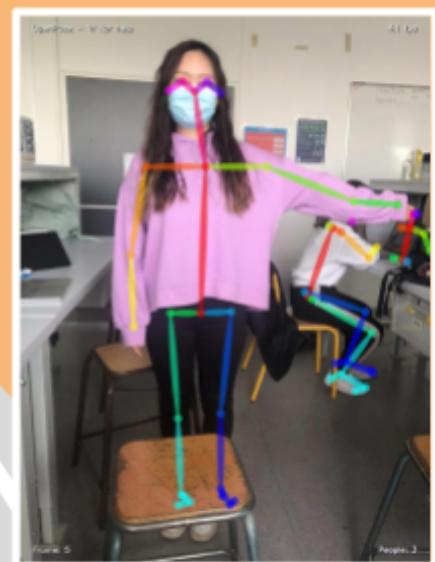
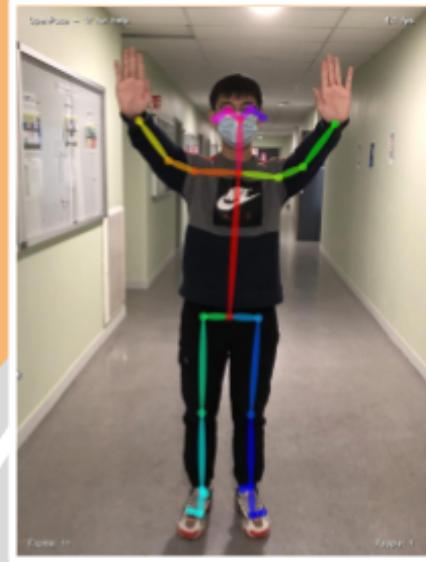


## Twizy vous facilite la vie!

Twizy autonome propose un service de conduite autonome qui est utile pour les handicapés.



Ce système autonome permet de réagir en analysant l'environnement (détecter les obstacles), suivre une personne à une vitesse réduite et comprendre la signe qu'elle donne.



# NOTICE D'UTILISATION

Guide d'installation et utilisation de notre système de conduite autonome

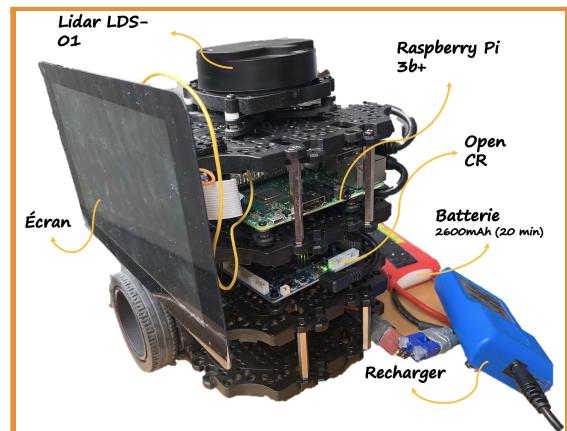


## DESCRIPTION DES ELEMENTS

L'objectif du notre projet industriel: Twizy autonome, est de programmer un robot capable de réaliser des tâches d'une voiture autonome. Plus précisément, le robot peut réagir en analysant l'environnement, suivre une personne à une vitesse réduite et comprendre le signe qu'elle donne. Il est composé d'un ensemble de composants permettant de traiter les images captées devant lui et déterminer s'il y a de personne qui donne le signe, scanner l'environnement autour de lui et donne une action après l'analyse. L'ensemble du système comprend les éléments suivants :

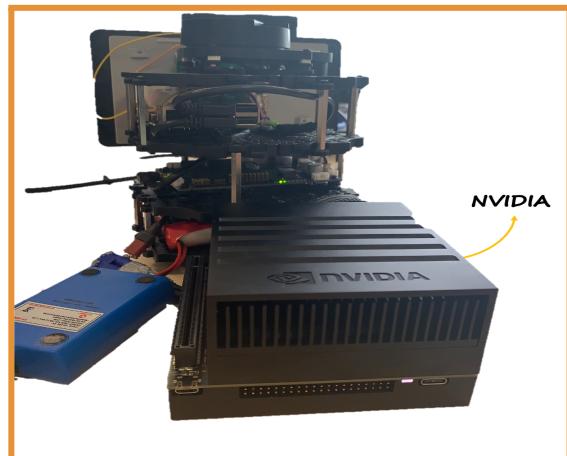
### - Turtlebot 3:

Il s'agit de l'unité principale du système. Il contient l'ensemble des modules réalisant le traitement de l'image et l'action demandée. Il est aussi le seul moyen d'interaction entre le système et l'utilisateur en utilisant le bouton ON et OFF qui se trouve dans Open CR.



On y retrouve les composants suivants :

- Lidar LDS-01
- Raspberry Pi 3B+
- Open CR
- Écran Raspberry Pi
- Batterie Li-Po
- Nvidia jetson AGX Xavier

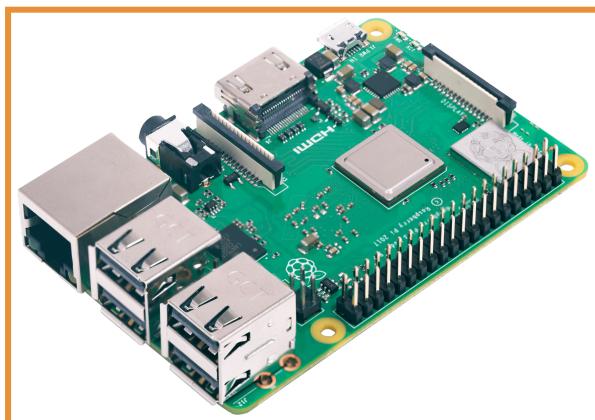


- **Lidar LDS-01:**

Ce capteur de distance laser 2D peut effectuer des mesures entre 12 et 350 cm de distance avec une grande précision. Il est monté sur une tête rotative qui lui permet de cartographier son environnement à 360° et se localiser. Il dispose d'un ROS(Robot Operating System) pour exploiter ses fonctionnalités. Il est relié avec le Raspberry Pi.



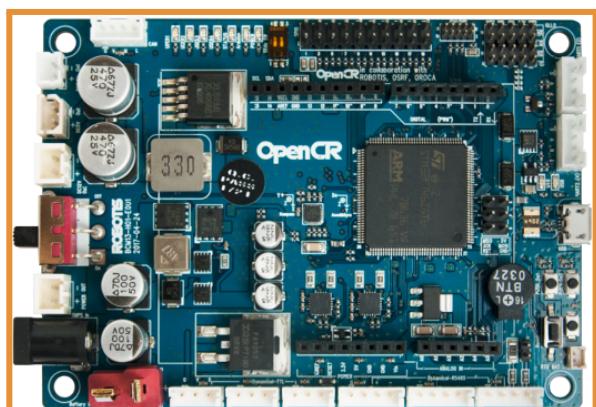
- **Raspberry Pi 3B+:**



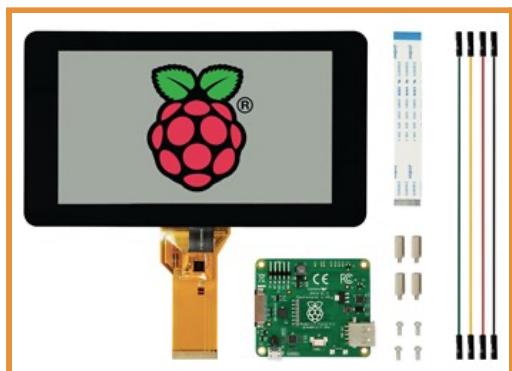
Il s'agit du module principal dans notre système. Il contient un processeur quad-core 64 bits avec 1,4 GHz, double bande de 2,4 GHz, LAN sans fil de 5 GHz et Bluetooth 4,2. On y implémente Raspbian OS qui est adapté avec la version Kinetic du Turtlebot 3.

- **Open CR:**

Il est développé pour le système embarqué ROS afin de fournir du matériel et des logiciels entièrement open-source. En plus, la puce de la série STM32F7 à l'intérieur de la carte Open CR est basée sur un ARM Cortex-M7. Dans le turtlebot 3, il sert à contrôler les roues du robot afin de lui permettre de tourner, avancer ou reculer.



- **Ecran Raspberry Pi:**



Cet écran est tactile. On connecte le ruban sur le port DSI de Raspberry-Pi. Et on l'alimente via le GPIO du Raspberry-Pi. Si on a besoin du clavier ou de la souris pour utiliser cet écran, on peut les connecter sur le port USB du Raspberry-Pi.

- **Batterie Li-Po:**

Le batterie a une capacité de 2600 mAh avec une tension d'alimentation de 11,1V. Il peut alimenter le système durant 20 minutes. Il est relié avec le openCR et le raspberry pi 3B+



- **Nvidia jetson AGX Xavier (GPU):**



Il s'agit du module qui traite les images qui sont captées dans l'environnement autour du robot. On y implémente Ubuntu 18.

## **UTILISATION DU SYSTÈME**

### **Mise en marche**

Pour marcher le système, mettre le bouton qui se situe sur le Open CR sur ON pour démarrer. Une led vert s'allume.

### **Fonctionnement de la structure du réseau sur GPU**

Il existe deux réseaux sur le GPU. Ce sont YOLO et Openpose.

Pour le modèle de réseau YOLO, vous devez d'abord entrer le dossier darknet et exécuter la commande suivante:

`./darknet detector demo cfg/coco.data cfg/yolov4.cfg yolov4.weights -c 0`

Le fichier de poids n'est pas unique, il existe de nombreuses options. Les besoins spécifiques doivent être sélectionnés en fonction des exigences de performances de calcul et de précision du réseau.

Après avoir exécuté cette commande, avec la caméra connectée, nous pouvons voir l'image en temps réel. Différents objets sont encadrés et marqués avec confiance. Nous avons apporté des modifications au code, et la personne avec la plus grande confiance sera interceptée et enregistrée en fonction de la "bounding box". La modification du chemin de sauvegarde des images nécessite la modification de la ligne 1059 de *darknet/src/image\_opencv.cpp*.

Pour plus d'informations, veuillez visiter ce site Web:

<https://github.com/AlexeyAB/darknet#how-to-use-on-the-command-line>

Pour le modèle de réseau Openpose, vous devez d'abord entrer le dossier openpose et exécuter la commande suivante:

- En exécutant la caméra:

`./build/examples/openpose/openpose.bin`

- Pour les photos:

`./build/examples/openpose/openpose.bin --image_dir examples/media/`

- Pour les vidéos:

`./build/examples/openpose/openpose.bin --video examples/media/video.avi`

En exécutant la commande ci-dessus, nous pouvons voir que les actions du personnage sont marquées. Les informations du nœud d'action de personnage seront enregistrées dans le fichier .json.

Pour plus d'informations, veuillez visiter ce site Web:

- [https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/doc/01\\_demo.md](https://github.com/CMU-Perceptual-Computing-Lab/openpose/blob/master/doc/01_demo.md)
- <https://github.com/CMU-Perceptual-Computing-Lab/openpose>

## Fonctionnement pour turtlebot 3

- Pour lancer le fonction de suivi du gen

Lancez ces commandes:

```
$ sudo apt-get install ros-kinetic-ar-track-alvar  
$ sudo apt-get install ros-kinetic-ar-track-alvar-msgs  
$ cd ~/catkin_ws/src  
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3_applications.git  
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3_applications_msgs.git  
$ cd ~/catkin_ws && catkin_make
```

Puis sur le PC distant, lancez:

```
$ sudo apt-get install python-pip  
$ sudo pip install -U scikit-learn numpy scipy  
$ sudo pip install --upgrade pip  
$ roscore
```

Puis sur le turtlebot 3, lancez:

```
$ roslaunch turtlebot3_bringup turtlebot3_robot.launch
```

Enfin sur le PC distant, lancez:

```
$ roslaunch turtlebot3_follow_filter turtlebot3_follow_filter.launch  
$ roslaunch turtlebot3_follower turtlebot3_follower.launch
```

- Pour lancer les code de scripts écrits par nous-même en utilisant LIDAR, il faut lancer ces commandes:

```
$ mkdir -p ~/wanderbot_ws/src  
$ cd ~/wanderbot_ws/src  
$ catkin_init_workspace  
$ cd ~/wanderbot_ws/src  
$ catkin_create_pkg wanderbot rospy geometry_msgs sensor_msgs  
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3_msgs.git  
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3.git  
$ git clone https://github.com/ROBOTIS-GIT/turtlebot3_simulations.git  
$ cd ~/wanderbot_ws  
$ catkin_make  
$ cd ~/wanderbot_ws/src/wanderbot/src
```

Alors on peut ajouter notre code de script et changer la permission et lancer le script:

```
$ chmod 777 xxx.py  
$ rosrun wanderbot xxx.py
```

Pour plus d'informations, veuillez visiter ce site Web:

<https://emanual.robotis.com/docs/en/platform/turtlebot3/quick-start/>

[http://alvarestech.com/temp/capp/GDT\\_Forma3D/Programming%20Robots%20with%20ROS%20by%20Morgan%20Quigley,%20Brian%20Gerkey,%20William%20D.%20Smart%20\(z-lib.org\).pdf](http://alvarestech.com/temp/capp/GDT_Forma3D/Programming%20Robots%20with%20ROS%20by%20Morgan%20Quigley,%20Brian%20Gerkey,%20William%20D.%20Smart%20(z-lib.org).pdf)

Pour télécharger le script qu'on a créé, veuillez visiter ce drive:

<https://drive.google.com/file/d/1OZytSLI4pRcPv1MSfaRUwC320w4CYfoy/view?usp=sharing>

## Maintenance

Si vous remarquez que la batterie se vide trop rapidement (< 20 min), il se peut que la batterie soit endommagée. Il faut peut-être la remplacer. Pour cela, procurez-vous une batterie supérieure ou égale à 2600 mAh avec la tension d'alimentation de 11,1V, éteignez le système (bouton sur OFF), débranchez la batterie actuelle et branchez la nouvelle. Allumez le système, vérifiez que la led verte s'allume. Si c'est le cas, le système fonctionne.

---

### Remarque:

- La version de Ubuntu dans le turtlebot 3 est 16, mais celle dans le Nvidia est 18, donc pour connecter le Nvidia dans le robot, il faut trouver une solution pour faire fonctionner les deux versions en parallèle sur un système.
- La tension d'alimentation pour le Nvidia n'est pas pareil que le batterie, pour résoudre ce problème, branchez deux batterie de 11,1V en parallèle.
- NVIDIA GPU AGX XAVIER n'a pas de module wifi. La connexion du GPU directement à l'ordinateur via un câble réseau ne partage pas le réseau avec le GPU, vous devez donc utiliser un routeur pour vous connecter via le câble réseau.
- Pour connecter le PC distant et turtlebot 3, il faut modifier l'adresse IP dans `~/.bashrc` dans les deux appareils.
- Lors du déploiement de la structure du réseau sur le GPU, vous devez faire attention à la version python. Sinon, l'opération peut échouer.
- **Identifiant du Nvidia:**

Username : twizy

Password : 123456

- **Identifiant du Turtlebot 3:**

Username : pi  
Password : 123456