FetchBot:
Build your intelligent Mars Rover

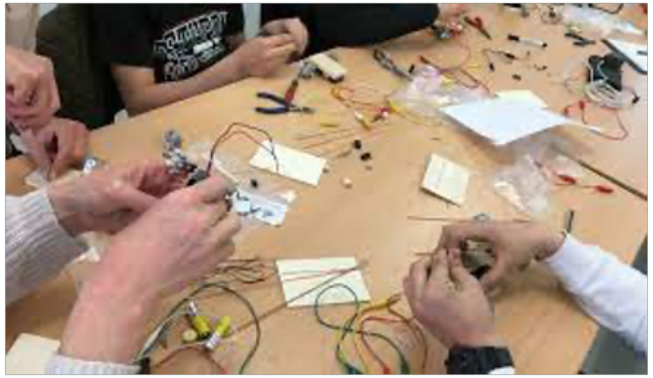
Yann-Aël Le Borgne, 26/08/2022

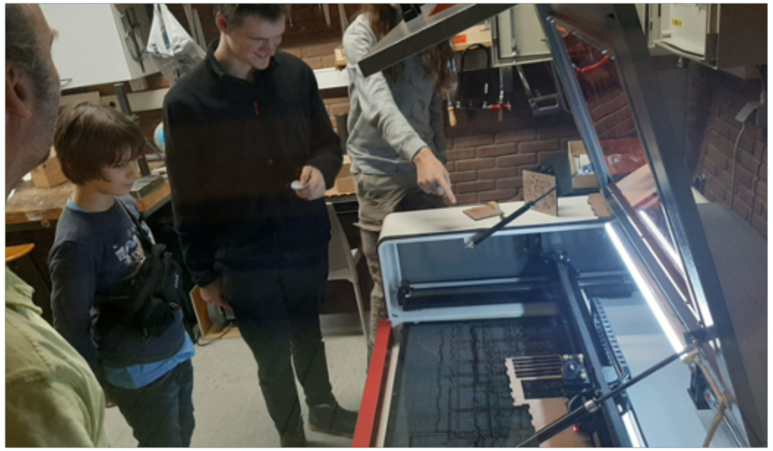
La Scientothèque and ESERO Belgium

La Scientothēque

L'ÉGALITÉ DES CHANCES PAR LES SCIENCES

Equal opportunities through sciences





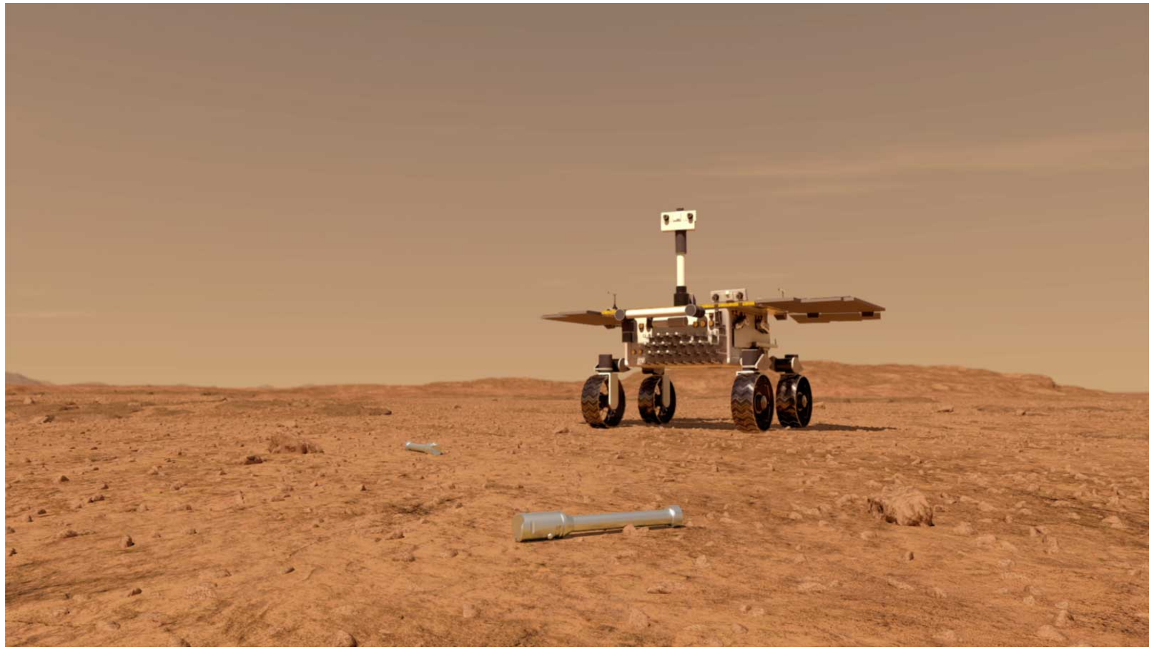
Belgium esero

Erasmus+

innoviris.brussels empowering research

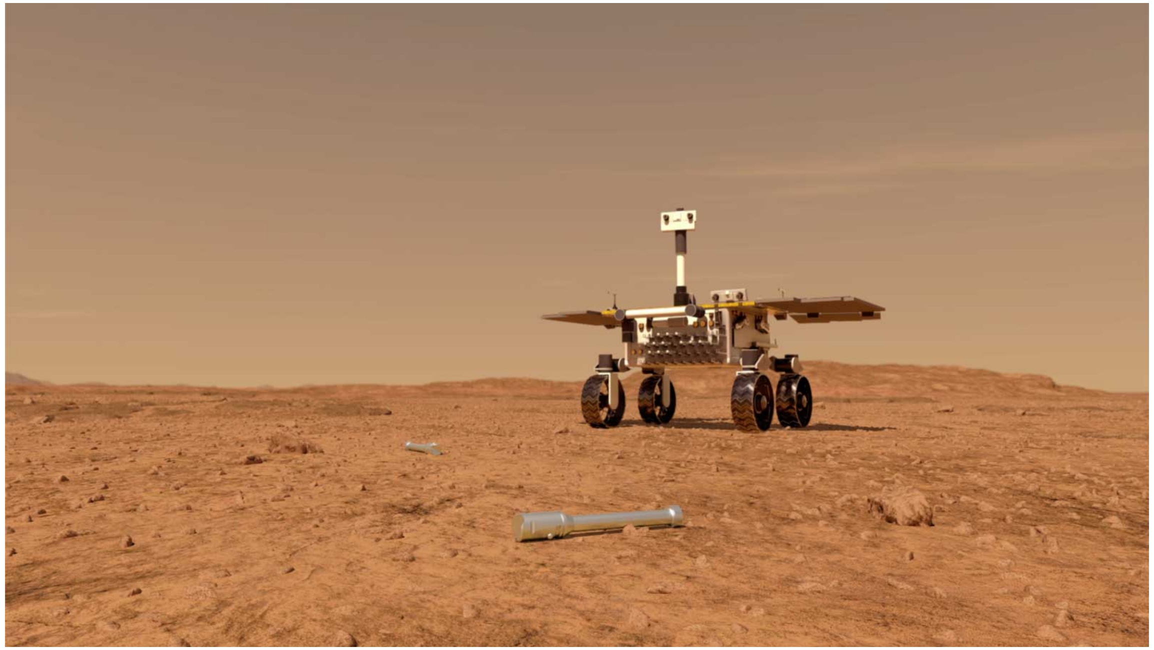
DIGITAL Belgium Skills Fund

FetchBot inspiration: Mars exploration - Fetch mission



1. Create an AI for image recognition and use it in a Scratch ou Python program
2. Build a Mars rover FetchBot
3. Program the FetchBot to find tubes on a Mars terrain

FetchBot inspiration: Mars exploration - Fetch mission



Constraints: Open source.
Financially affordable.
Simple to build.
Programming in Scratch (10-14 ans) or Python (14-18 ans).

Image recognition

Train a model to detect tubes with the Teachable Machine

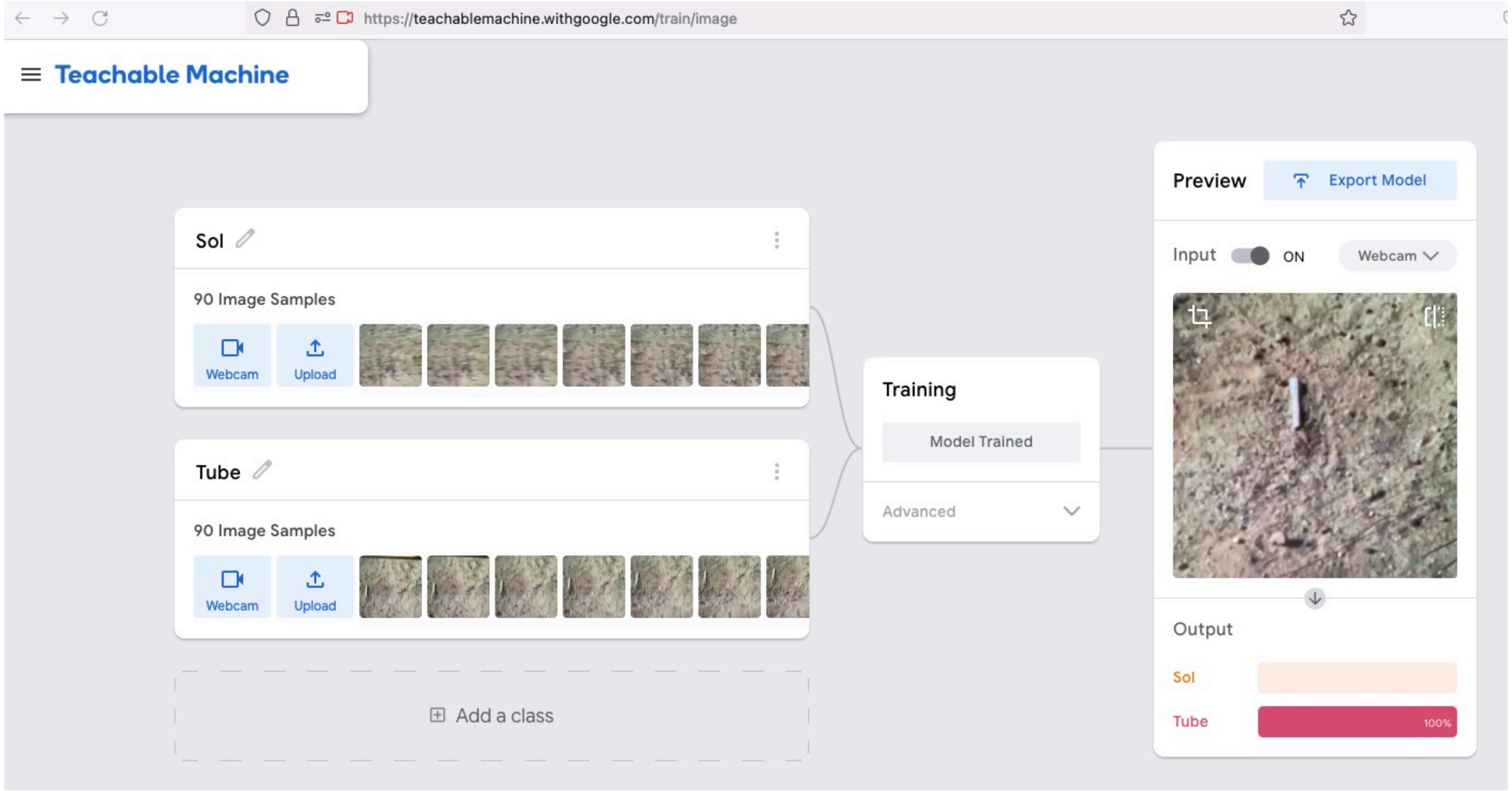
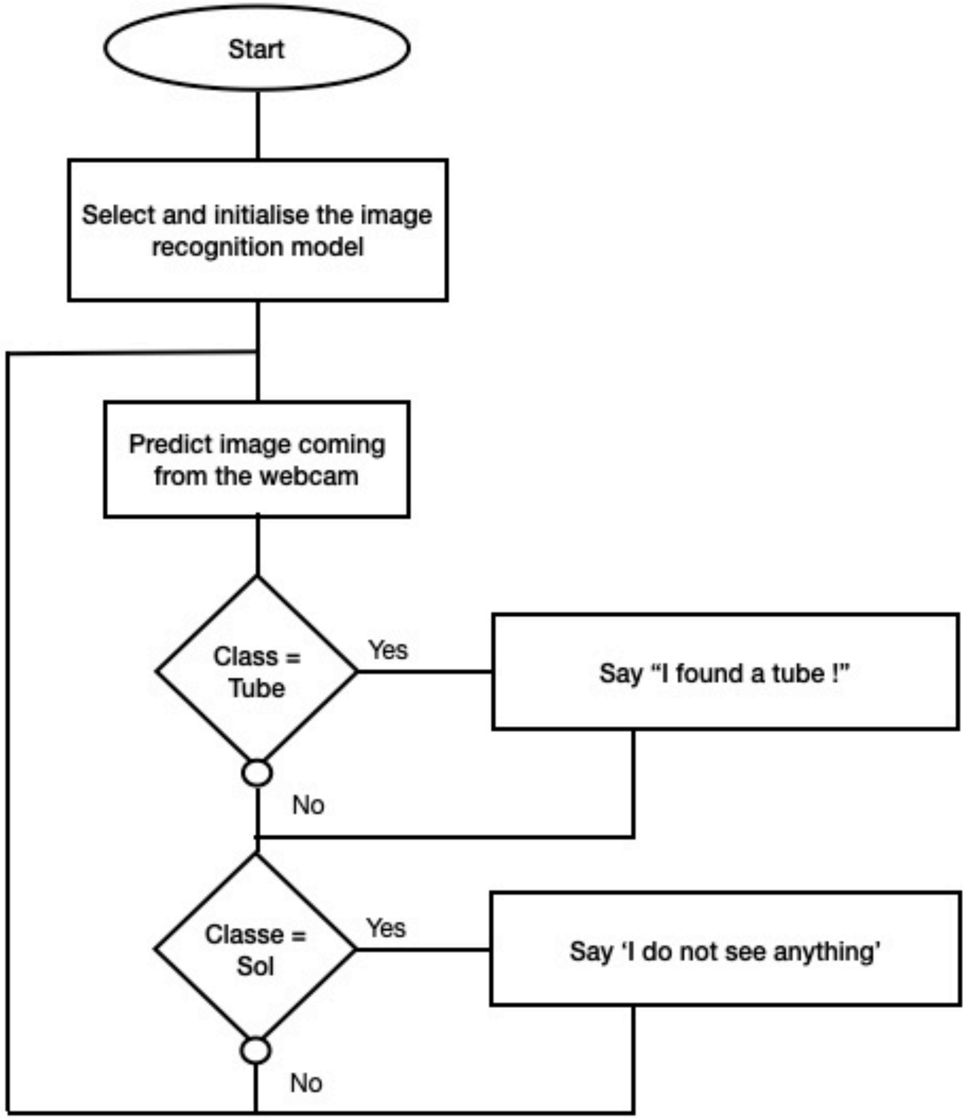
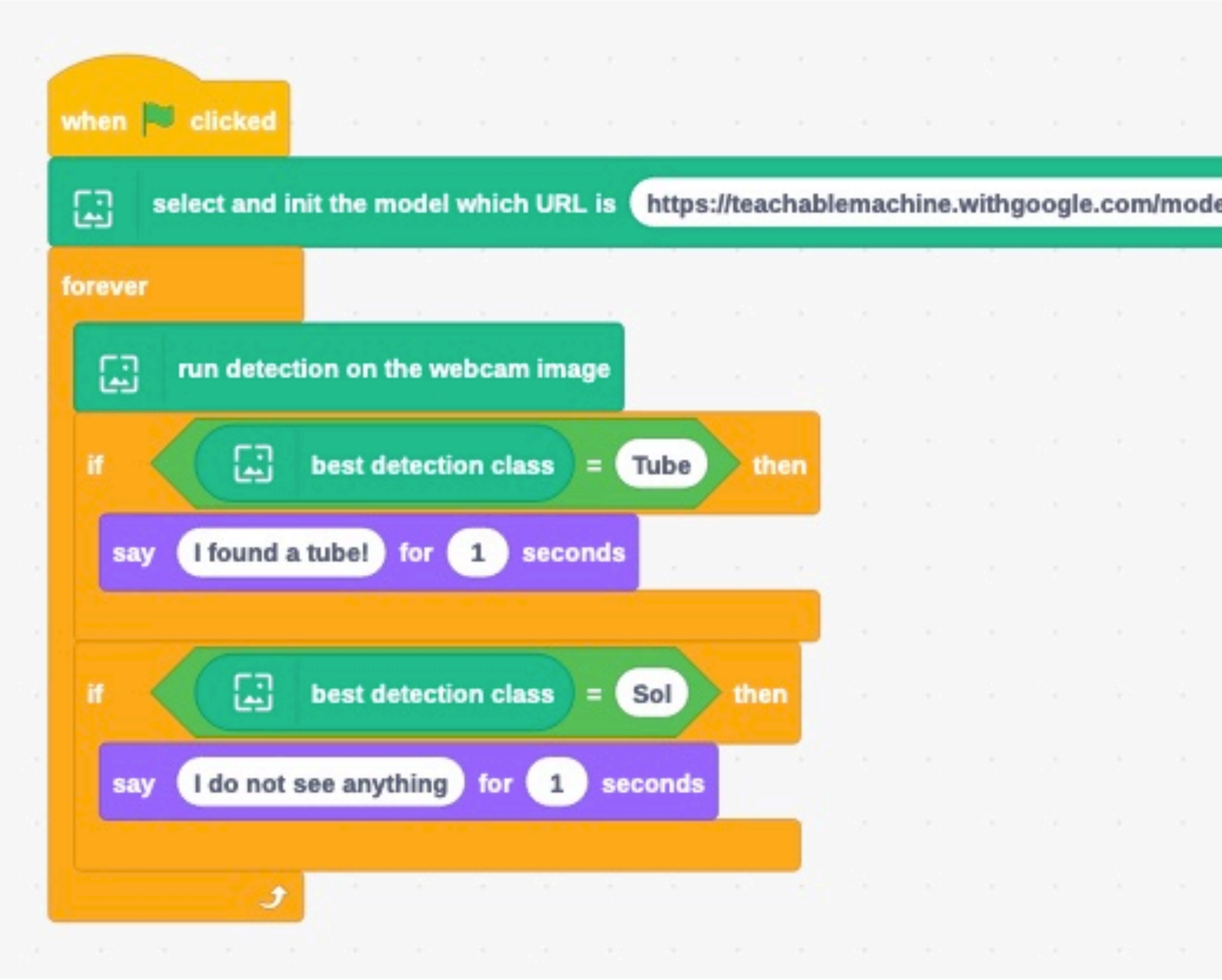


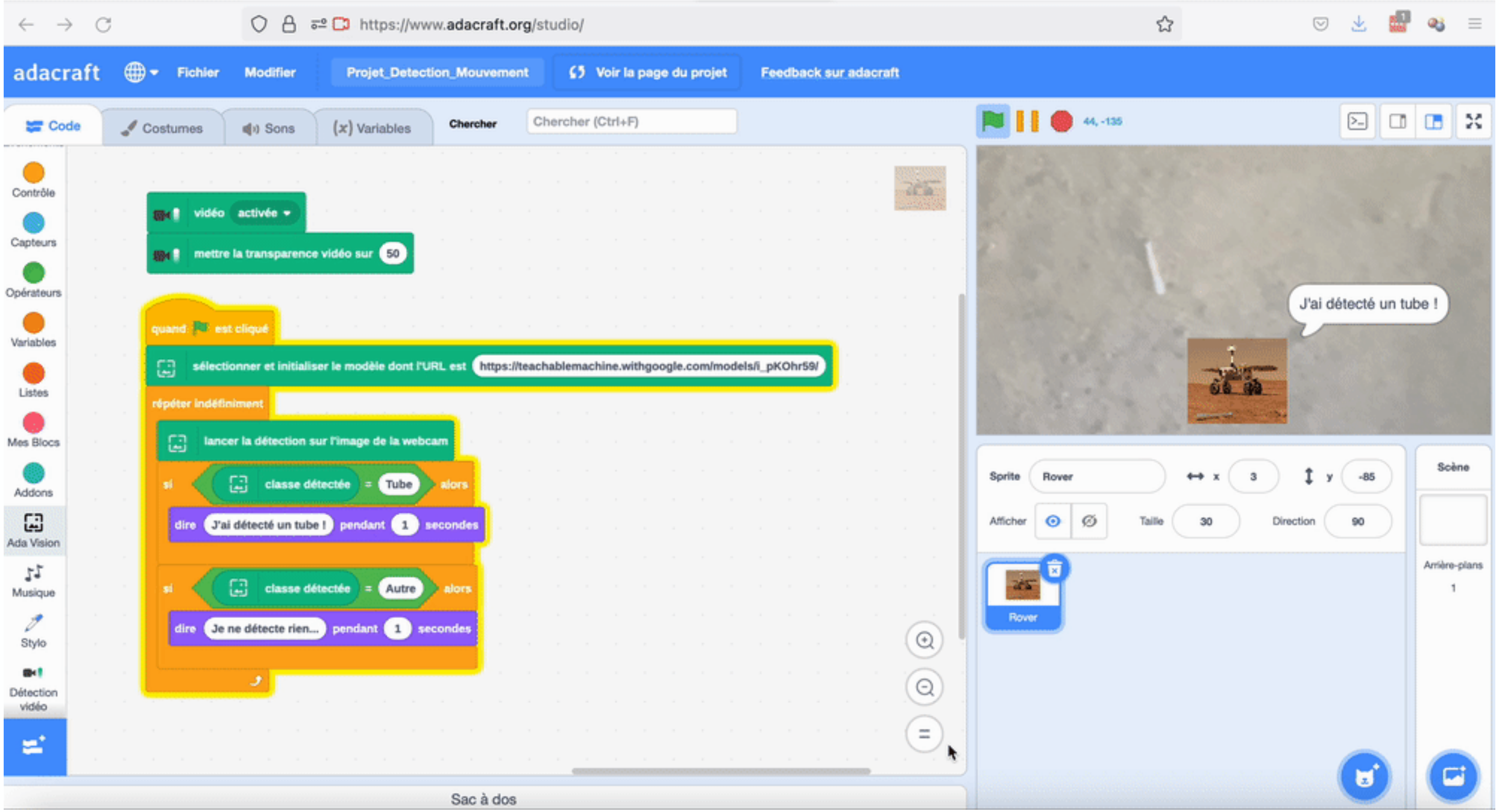
Image recognition with Scratch (Adacraft)


Start
Select and initialise the image recognition model
Predict image coming from the webcam
Class = Tube
Yes
Say "I found a tube !"
No
Classe = Sol
Yes
Say 'I do not see anything'
No



```
when 🏳 clicked
select and init the model which URL is (https://teachablemachine.withgoogle.com/mode
forever
    run detection on the webcam image
    if <(best detection class) = (Tube)> then
        say (I found a tube!) for (1) seconds
    if <(best detection class) = (Sol)> then
        say (I do not see anything) for (1) seconds
```

Image recognition with Scratch (Adacraft)



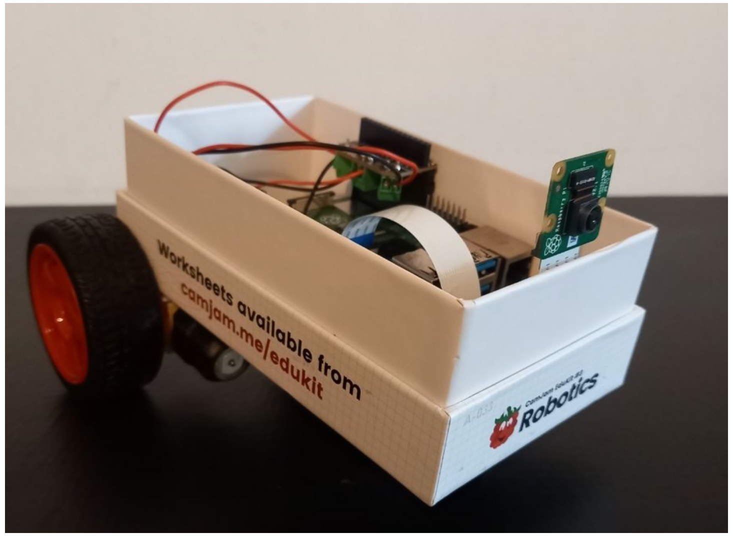
https://www.adacraft.org/studio/

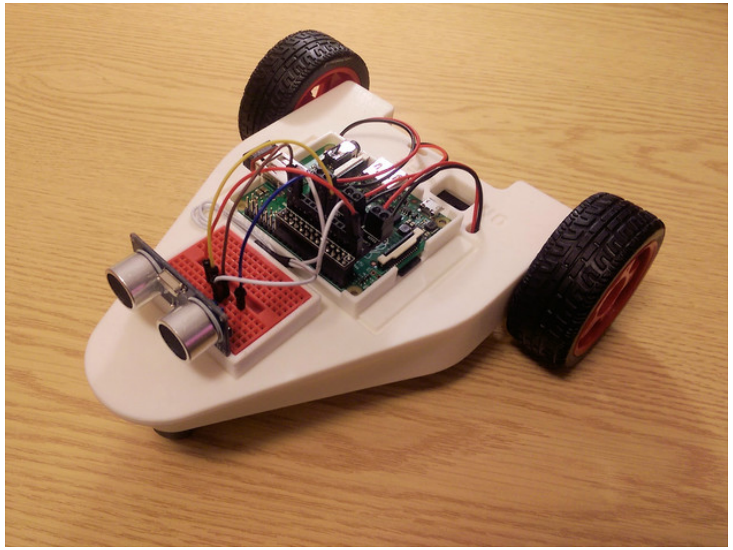
```
vidéo activée
mettre la transparence vidéo sur 50

quand 🏴 est cliqué
sélectionner et initialiser le modèle dont l'URL est https://teachablemachine.withgoogle.com/models/i_pKOhr59/
répéter indéfiniment
    lancer la détection sur l'image de la webcam
    si classe détectée = Tube alors
        dire J'ai détecté un tube ! pendant 1 secondes
    si classe détectée = Autre alors
        dire Je ne détecte rien... pendant 1 secondes
```

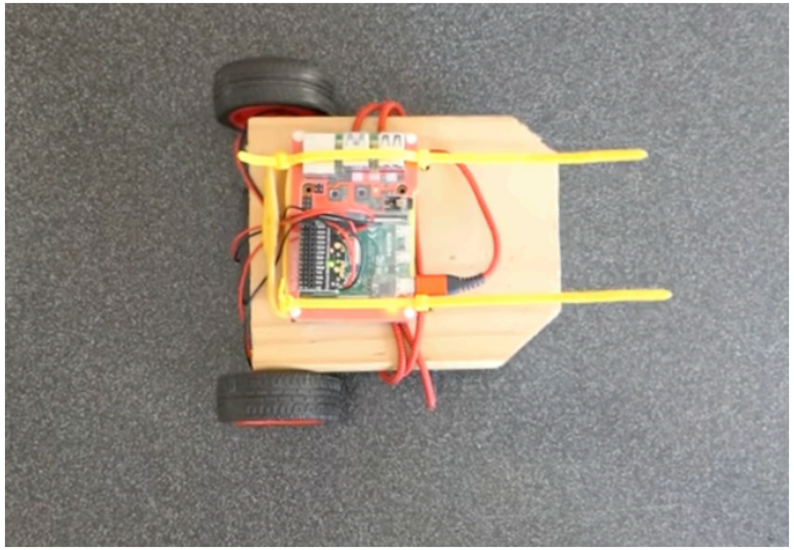
Sprite: Rover · x: 3 · y: -85 · Taille: 30 · Direction: 90

FetchBot: Hardware

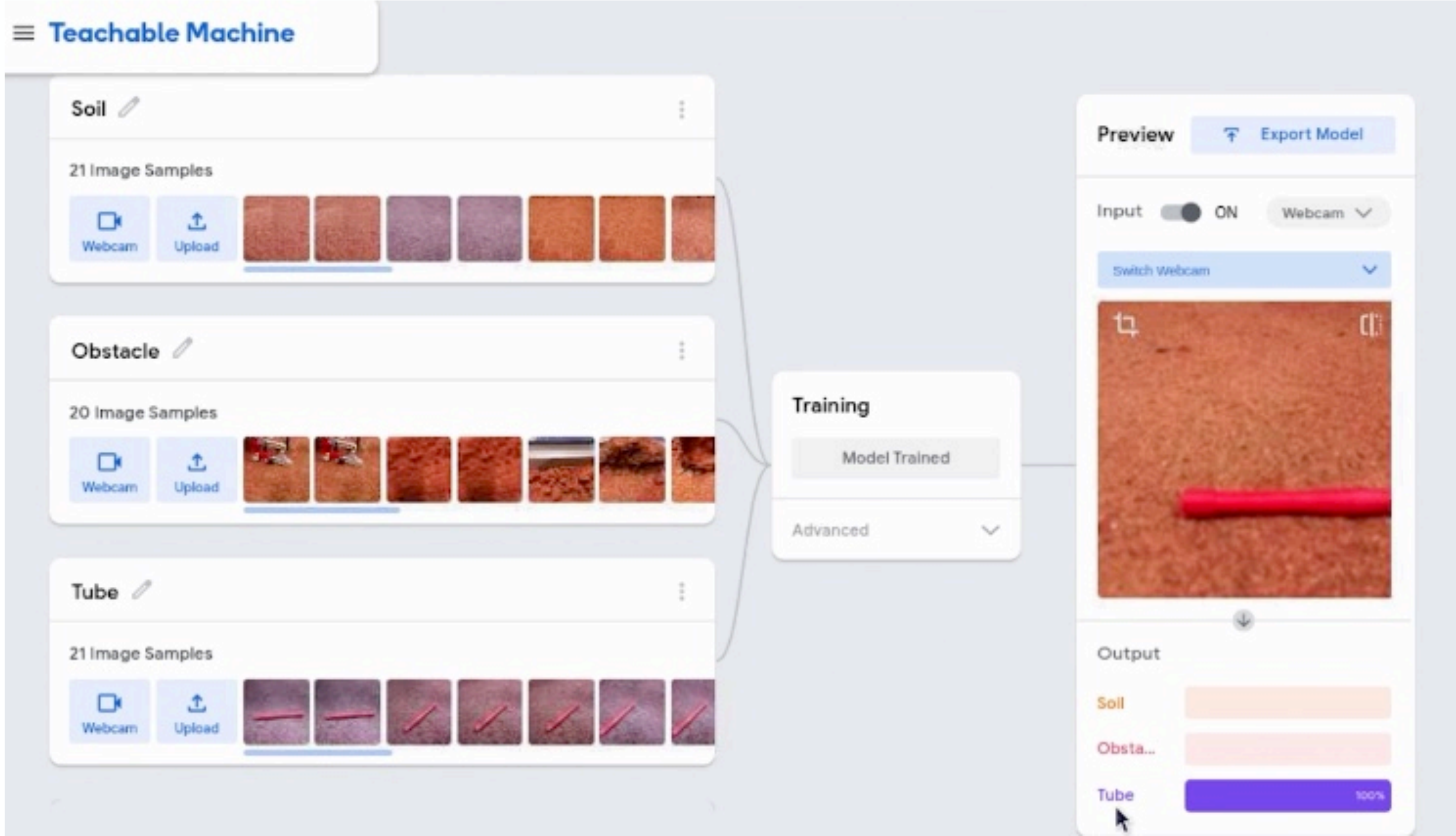
- Raspberry 3 or 4, and camera (AstroPi Kit)
- Rover: CamJam EduKit (Around 20 euros)







Training the rover to avoid obstacles and find tubes



Teachable Machine

Soil

21 Image Samples

Webcam Upload

Obstacle

20 Image Samples

Webcam Upload

Tube

21 Image Samples

Webcam Upload

Training

Model Trained

Advanced

Preview Export Model

Input ON Webcam

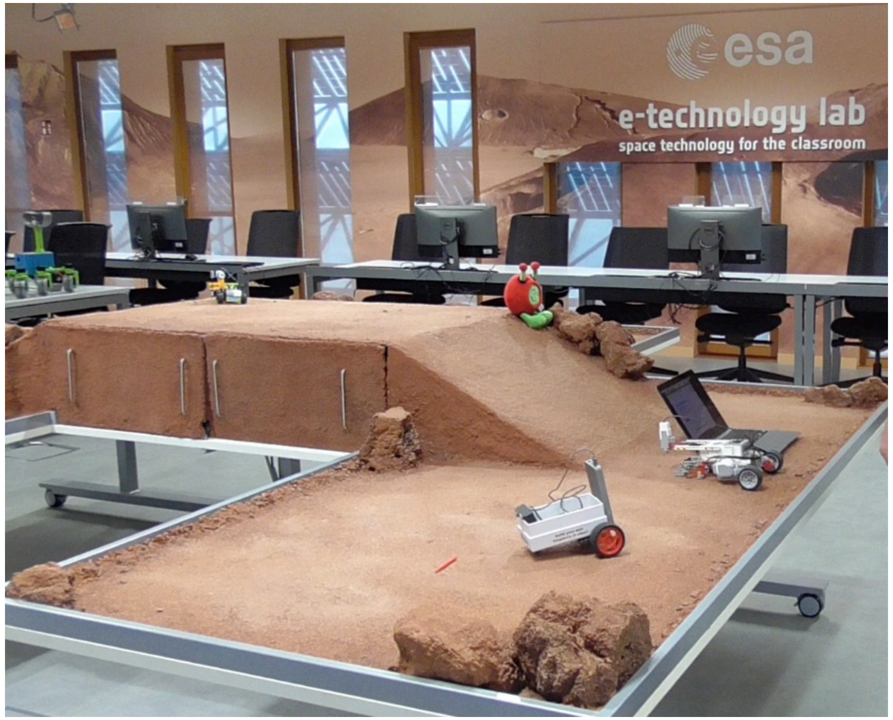
Switch Webcam

Output

Soil

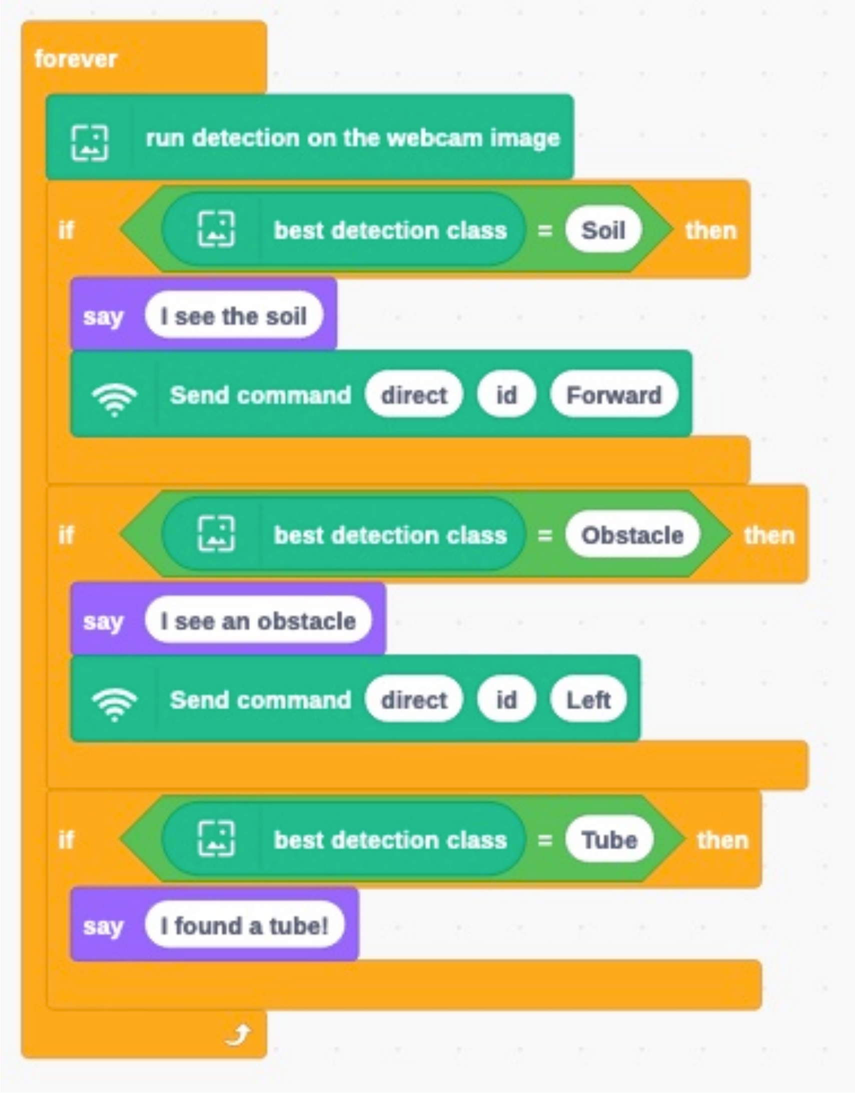
Obsta...

Tube 100%



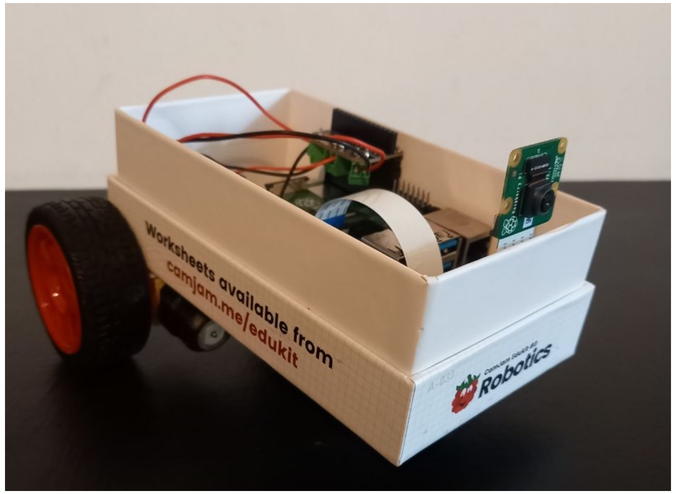
Control the rover to avoid obstacles and find tubes

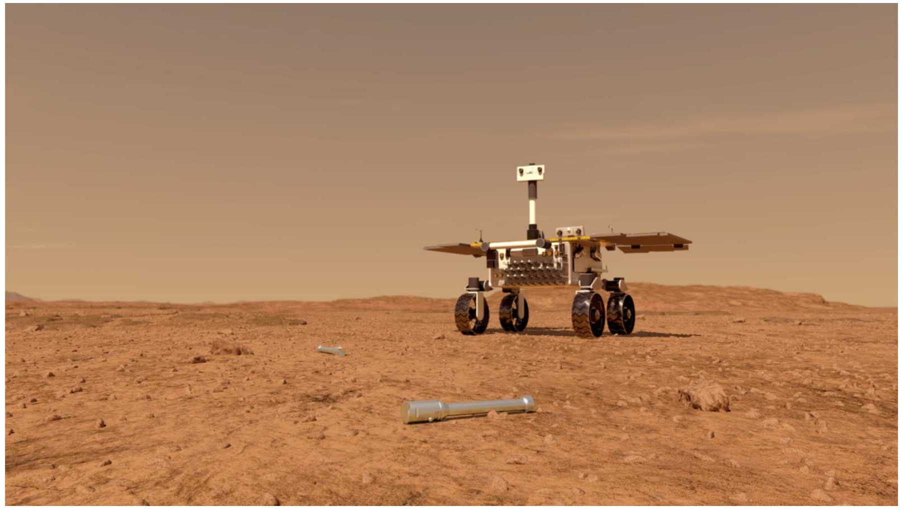
- If class is ‘Soil’ then move forward
- If class is ‘Obstacle’ then turn left
- If class is ‘Tube’ then say ‘I found a tube!’



```
forever
    run detection on the webcam image
    if best detection class = Soil then
        say I see the soil
        Send command direct id Forward
    if best detection class = Obstacle then
        say I see an obstacle
        Send command direct id Left
    if best detection class = Tube then
        say I found a tube!
```

FetchBot: Demo

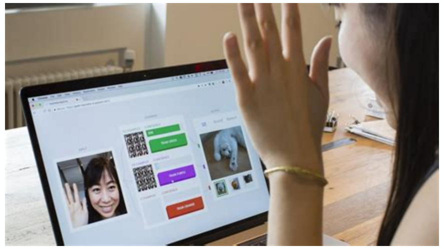




FetchBot project page :

https://lascientotheque.github.io/fetchbot-fr

Summary and extensions



Teachable Machine

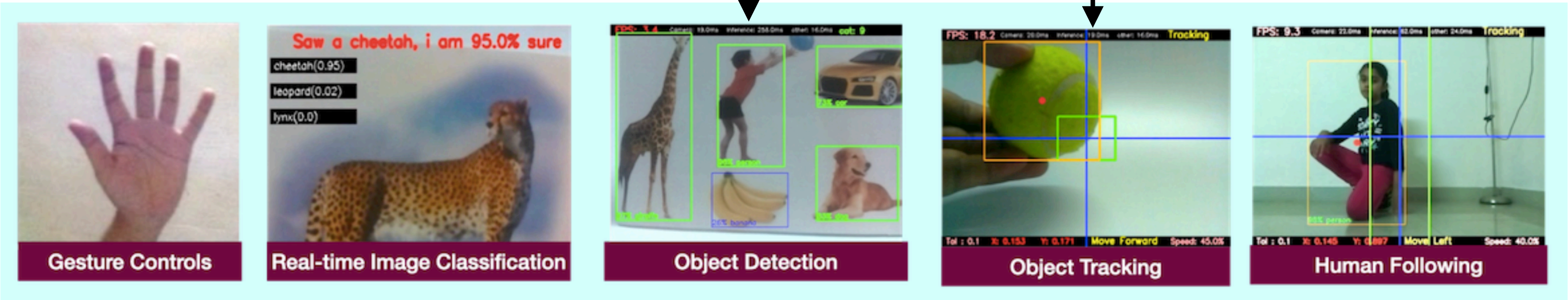
Export model

10-14

14-18

Implementation

Applications



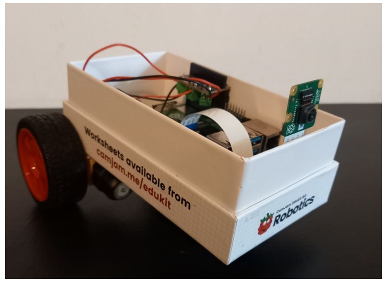
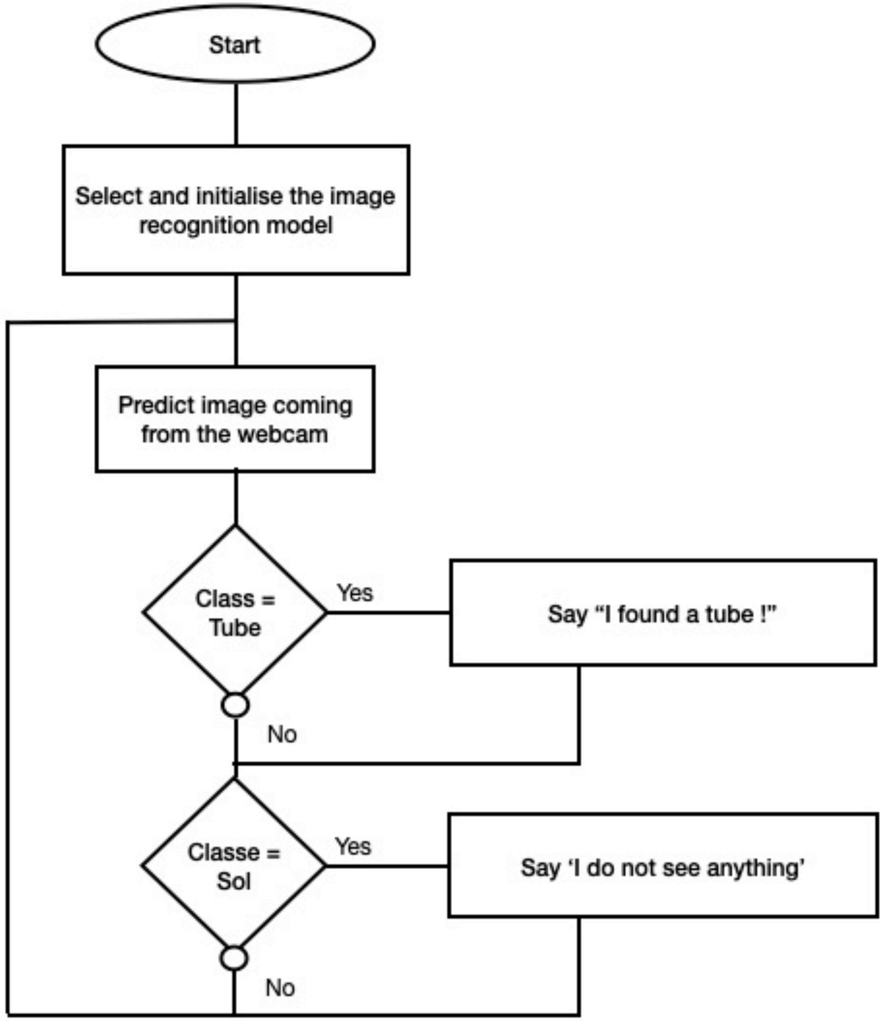
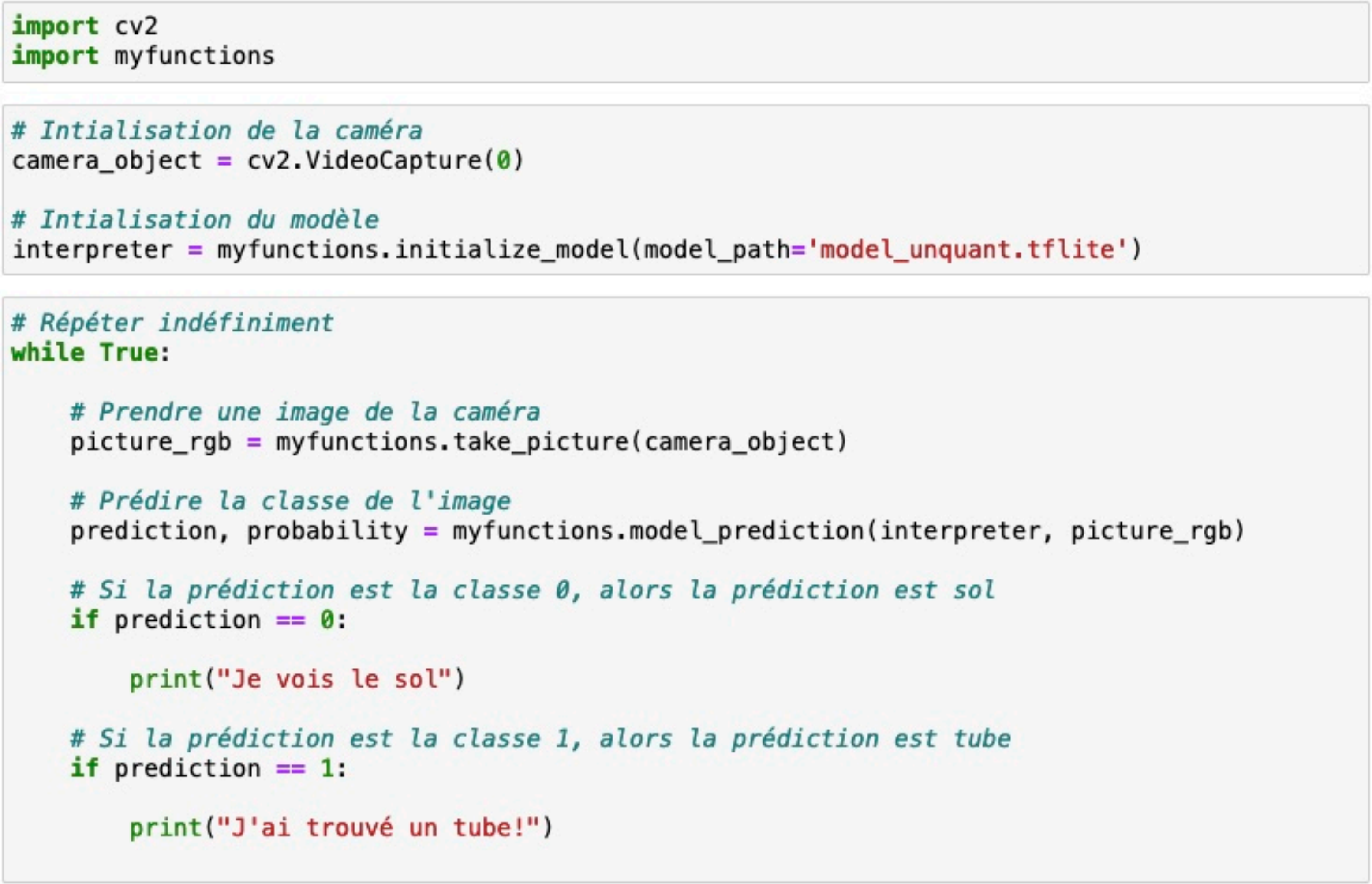


Image recognition with Python


Start
Select and initialise the image recognition model
Predict image coming from the webcam
Class = Tube
Yes
Say "I found a tube !"
No
Classe = Sol
Yes
Say 'I do not see anything'
No



```
import cv2
import myfunctions
```

```
# Intialisation de la caméra
camera_object = cv2.VideoCapture(0)

# Intialisation du modèle
interpreter = myfunctions.initialize_model(model_path='model_unquant.tflite')
```

```
# Répéter indéfiniment
while True:

    # Prendre une image de la caméra
    picture_rgb = myfunctions.take_picture(camera_object)

    # Prédire la classe de l'image
    prediction, probability = myfunctions.model_prediction(interpreter, picture_rgb)

    # Si la prédiction est la classe 0, alors la prédiction est sol
    if prediction == 0:

        print("Je vois le sol")

    # Si la prédiction est la classe 1, alors la prédiction est tube
    if prediction == 1:

        print("J'ai trouvé un tube!")
```