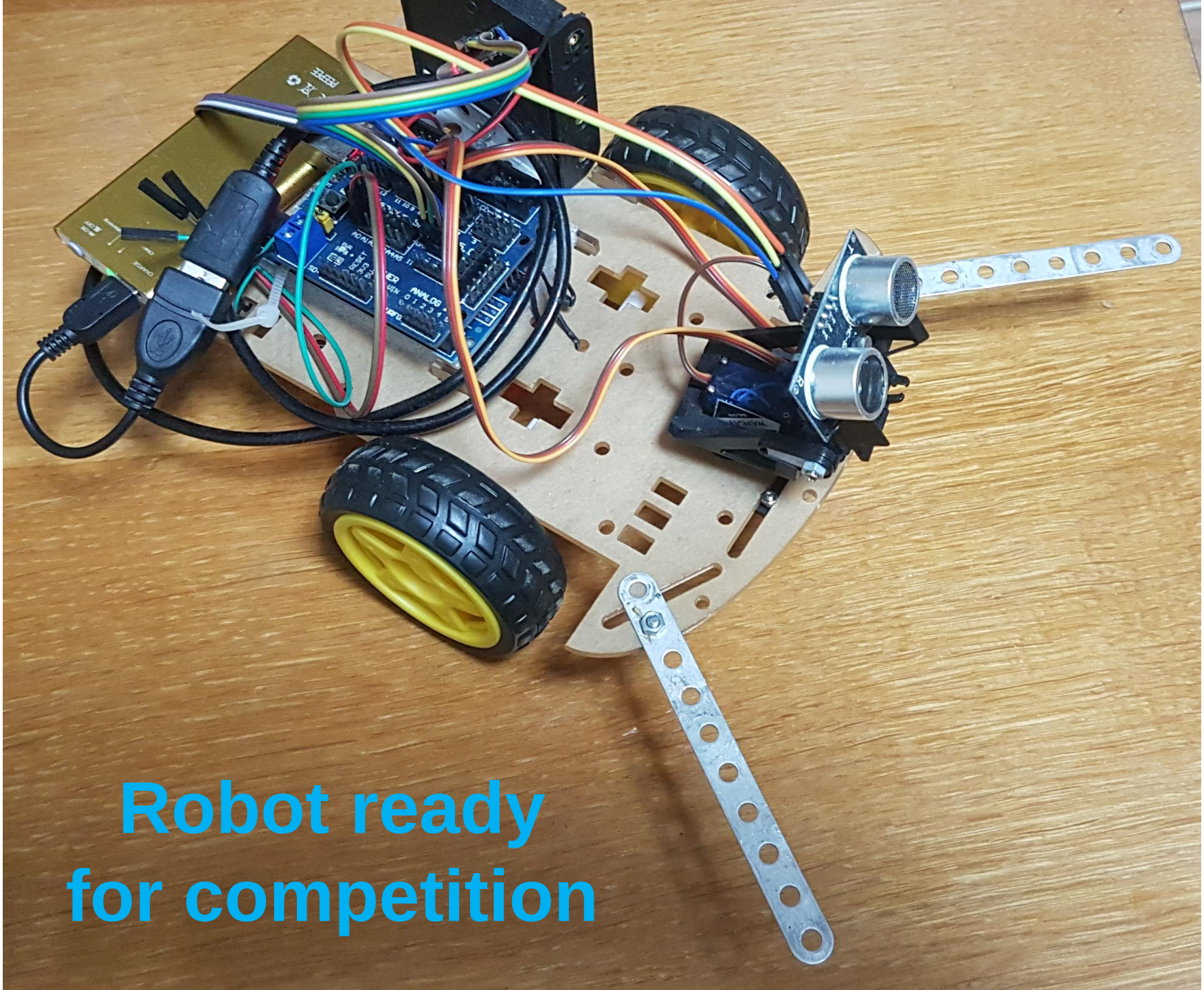
International Conference "Innovation Development in the Industry 4.0 Paradigm"

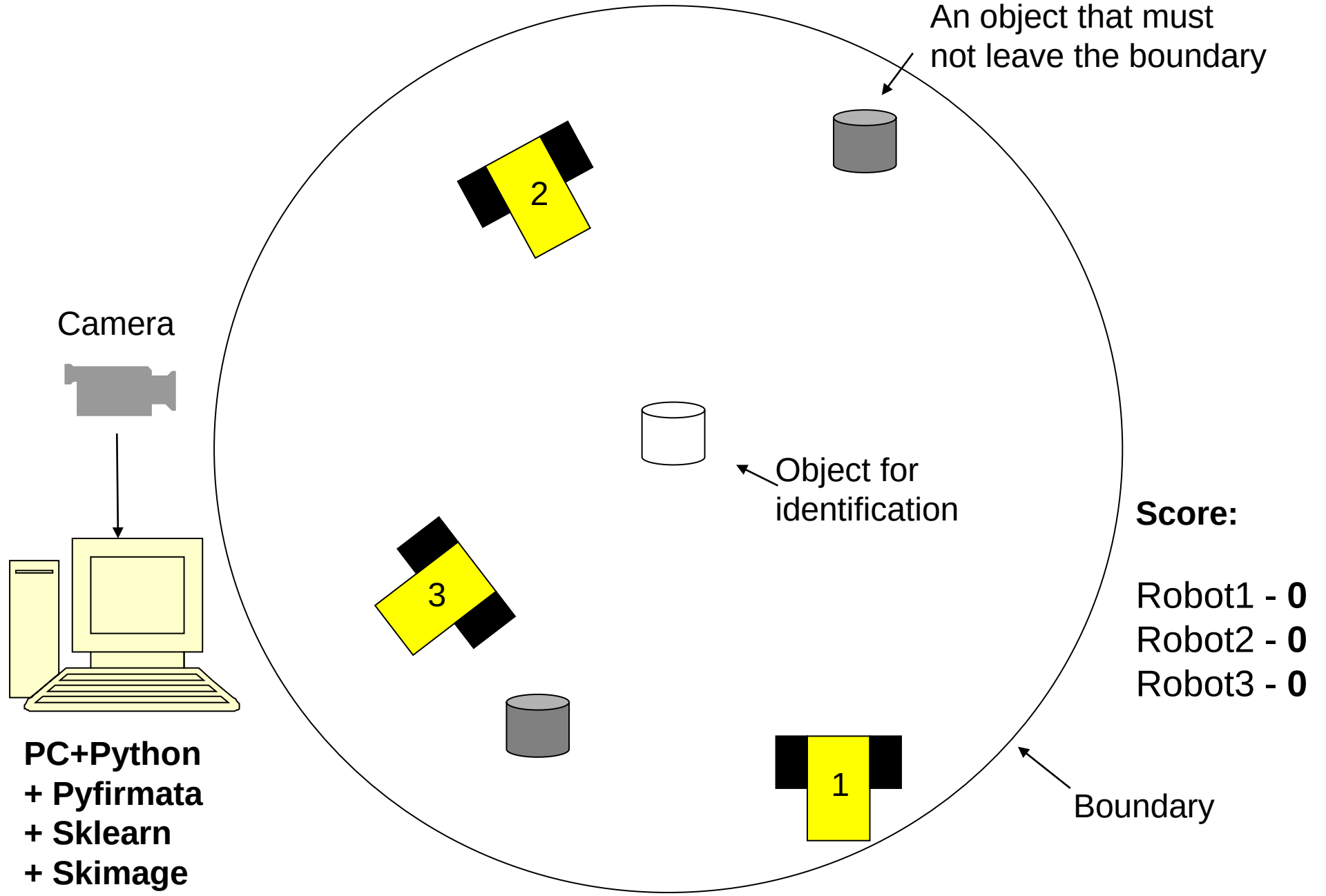
ROBOT COMPETITION FRAMEWORK BASED ON ARDUINO, PYTHON, OPENCV AND SCIKIT-LEARN

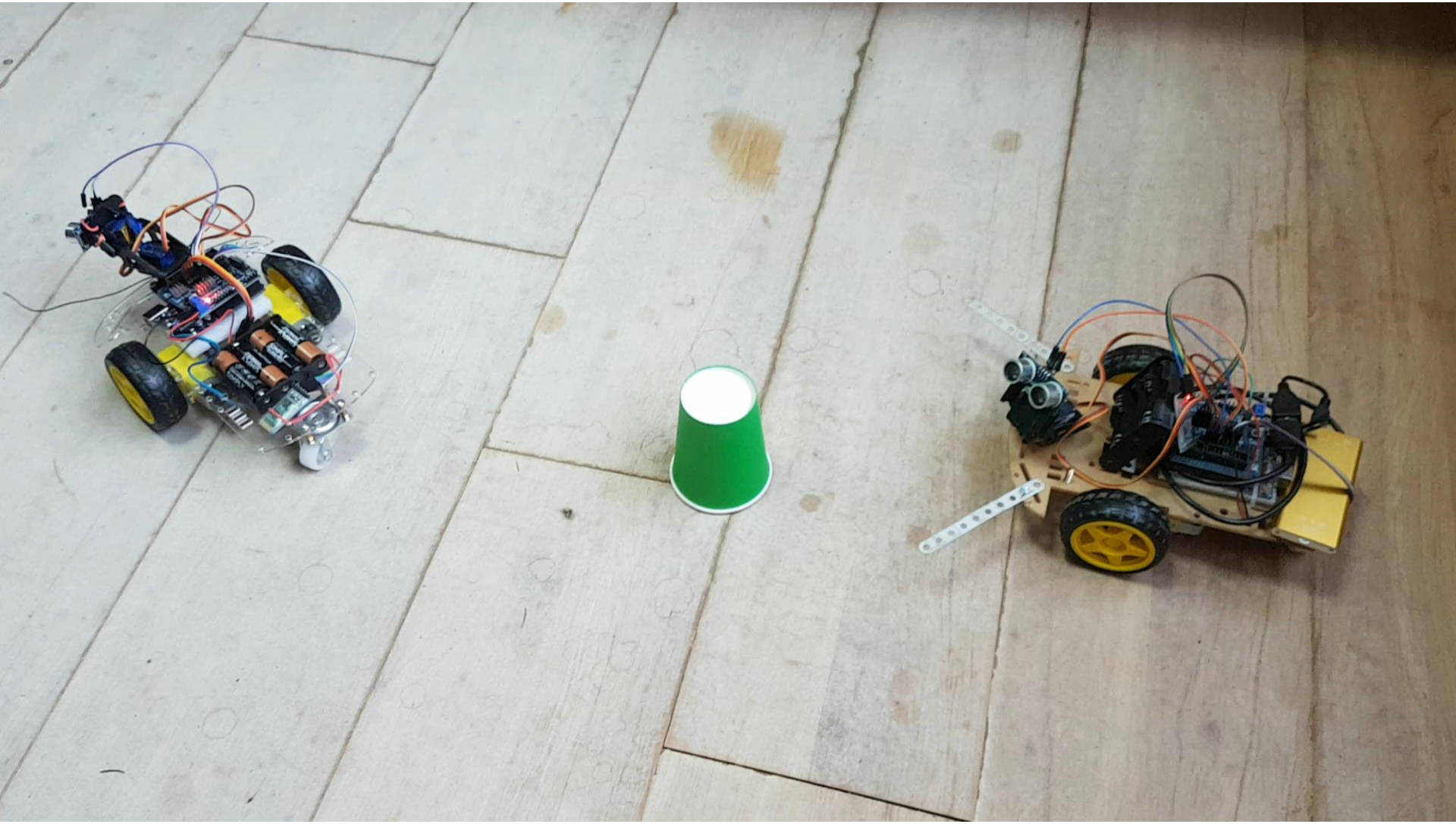
Volodymyr Kopei, Ihor Proniuk

Ivano-Frankivsk National Technical University of Oil and Gas
email: vkopey@gmail.com


Robot ready
for competition

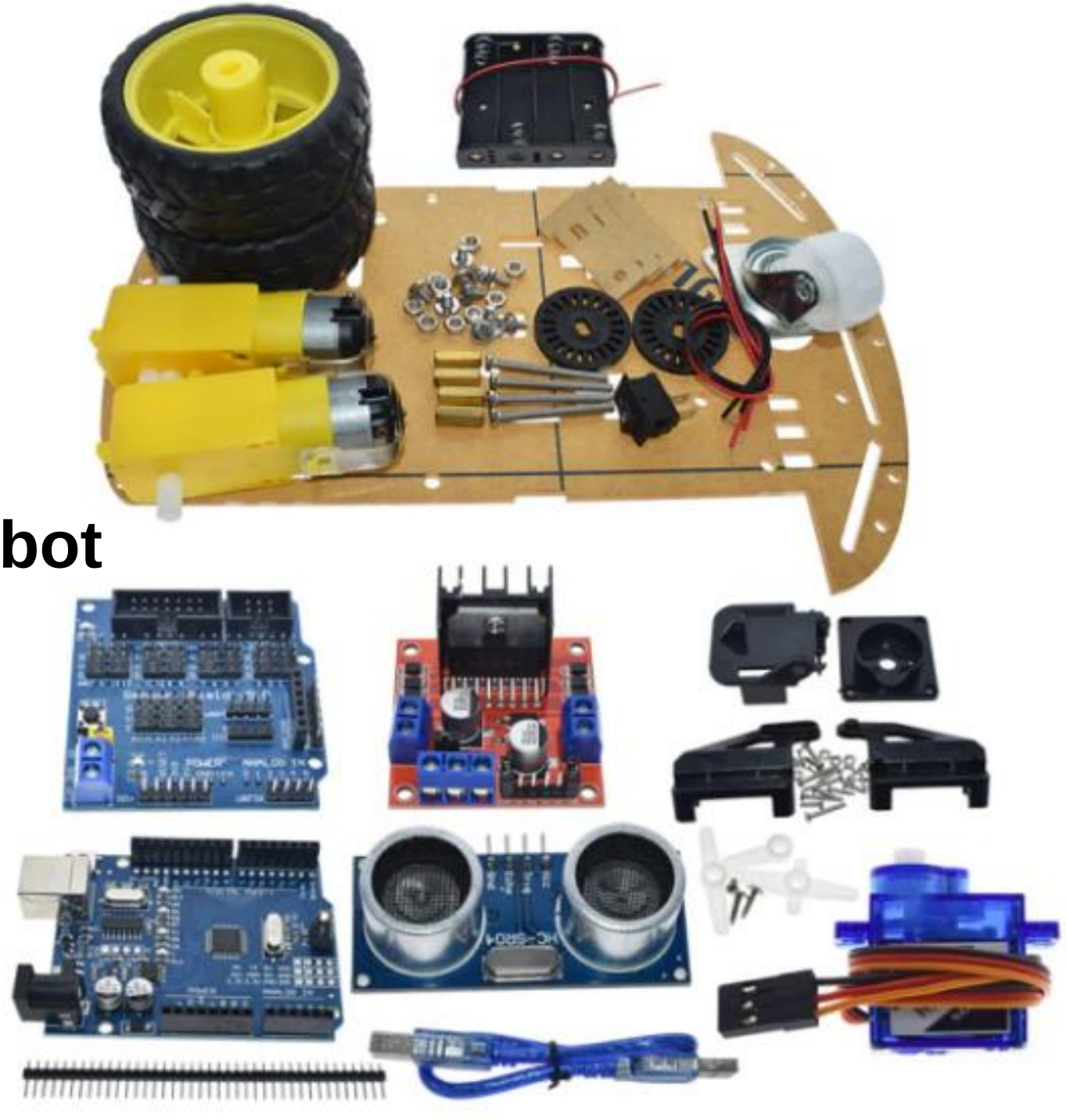
Robot competition



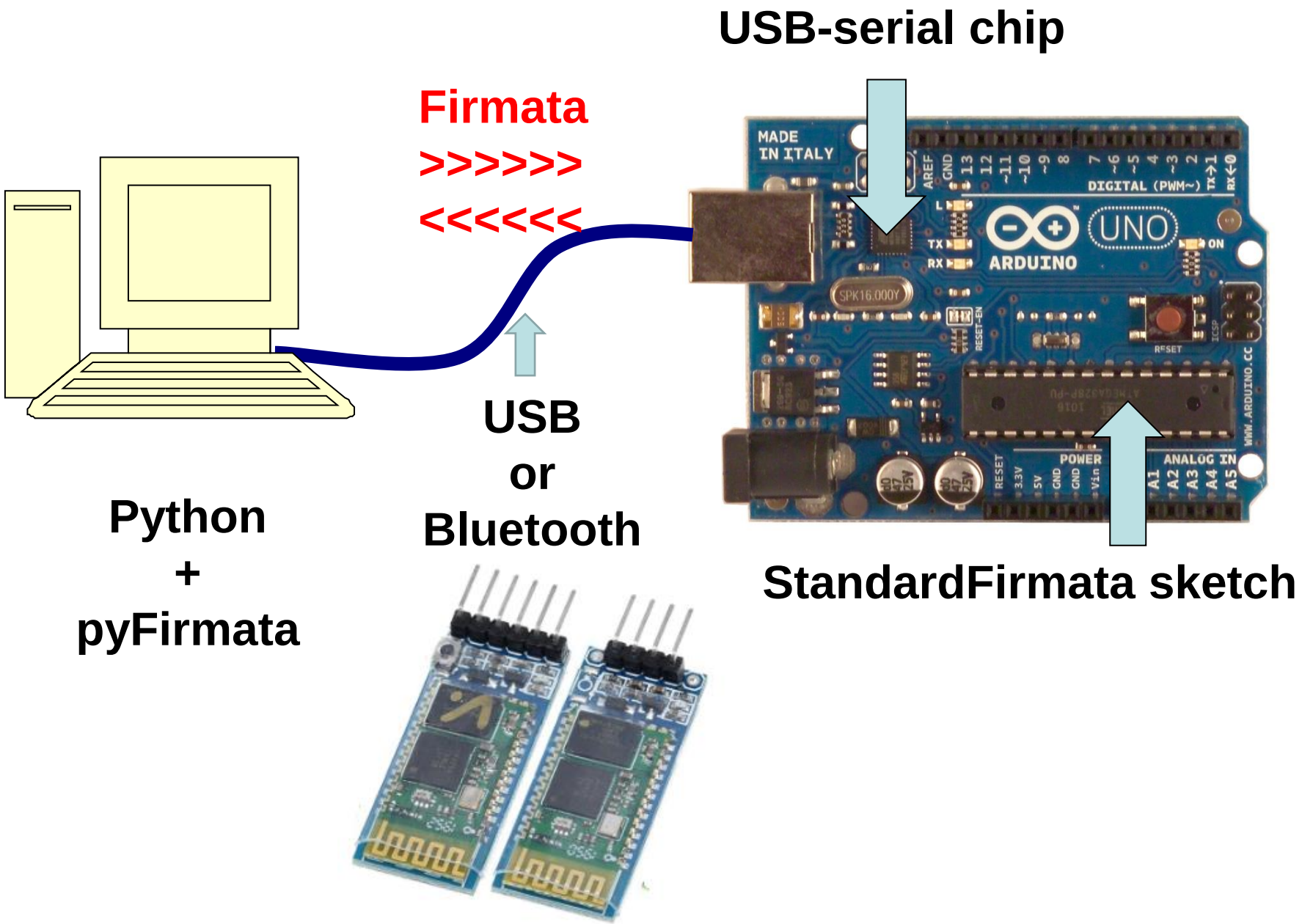


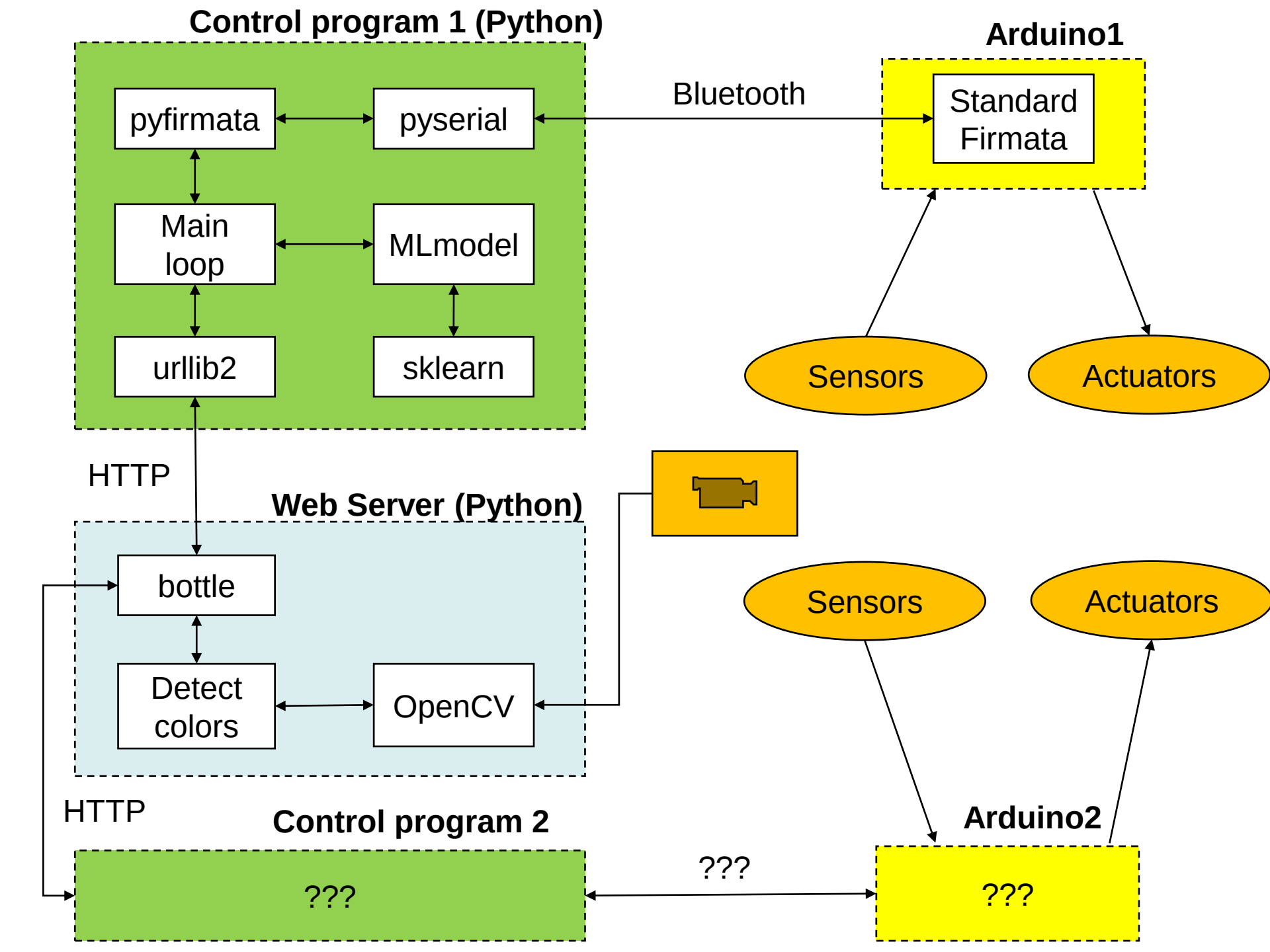
Basic kit for building a robot

~20 $

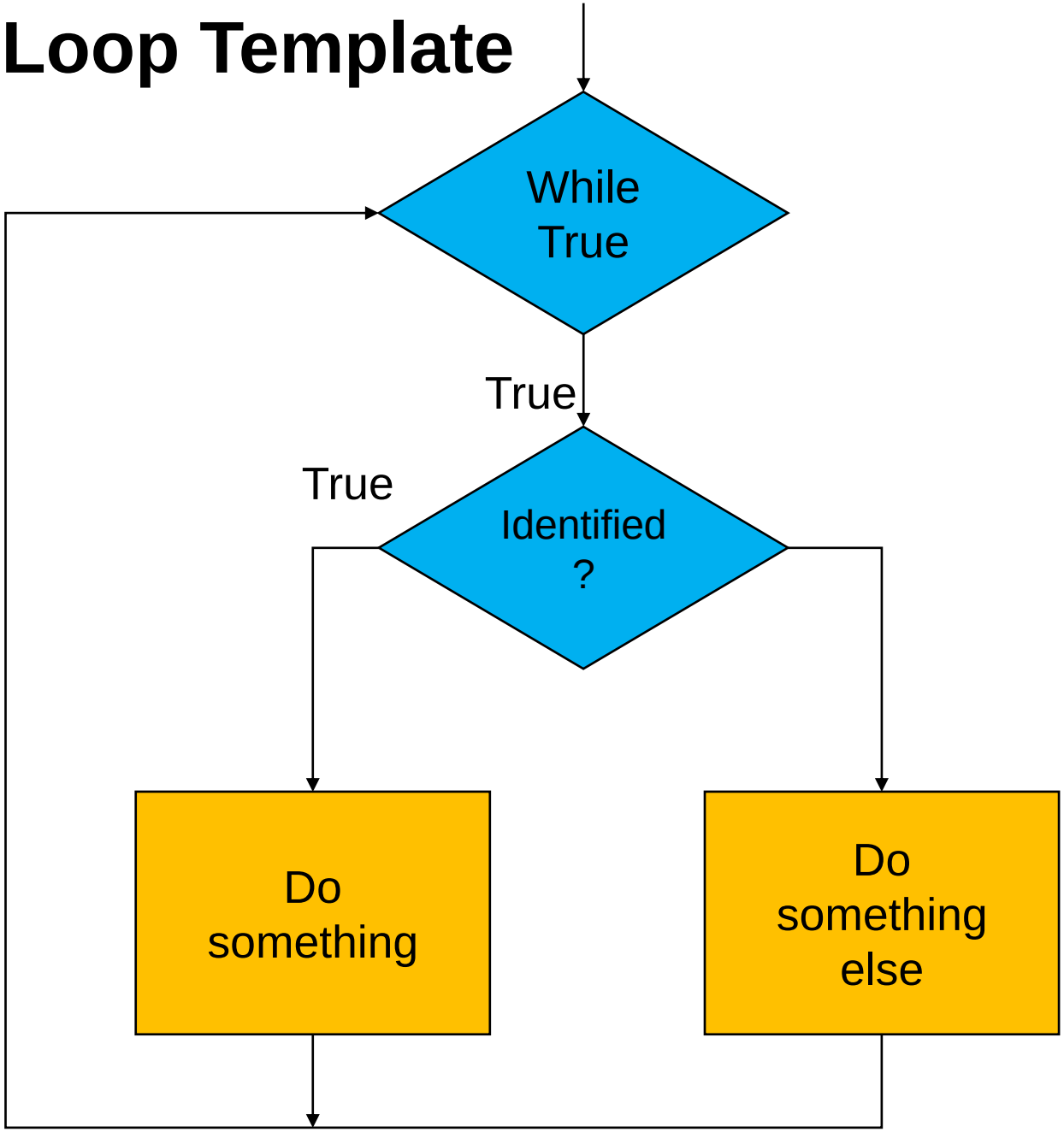


PC-Arduino communication




Control program 1 (Python)
pyfirmata
pyserial
Main loop
MLmodel
urllib2
sklearn
Bluetooth
Arduino1
Standard Firmata
Sensors
Actuators
HTTP
Web Server (Python)
bottle
Detect colors
OpenCV
Sensors
Actuators
HTTP
Control program 2
???
???
Arduino2
???

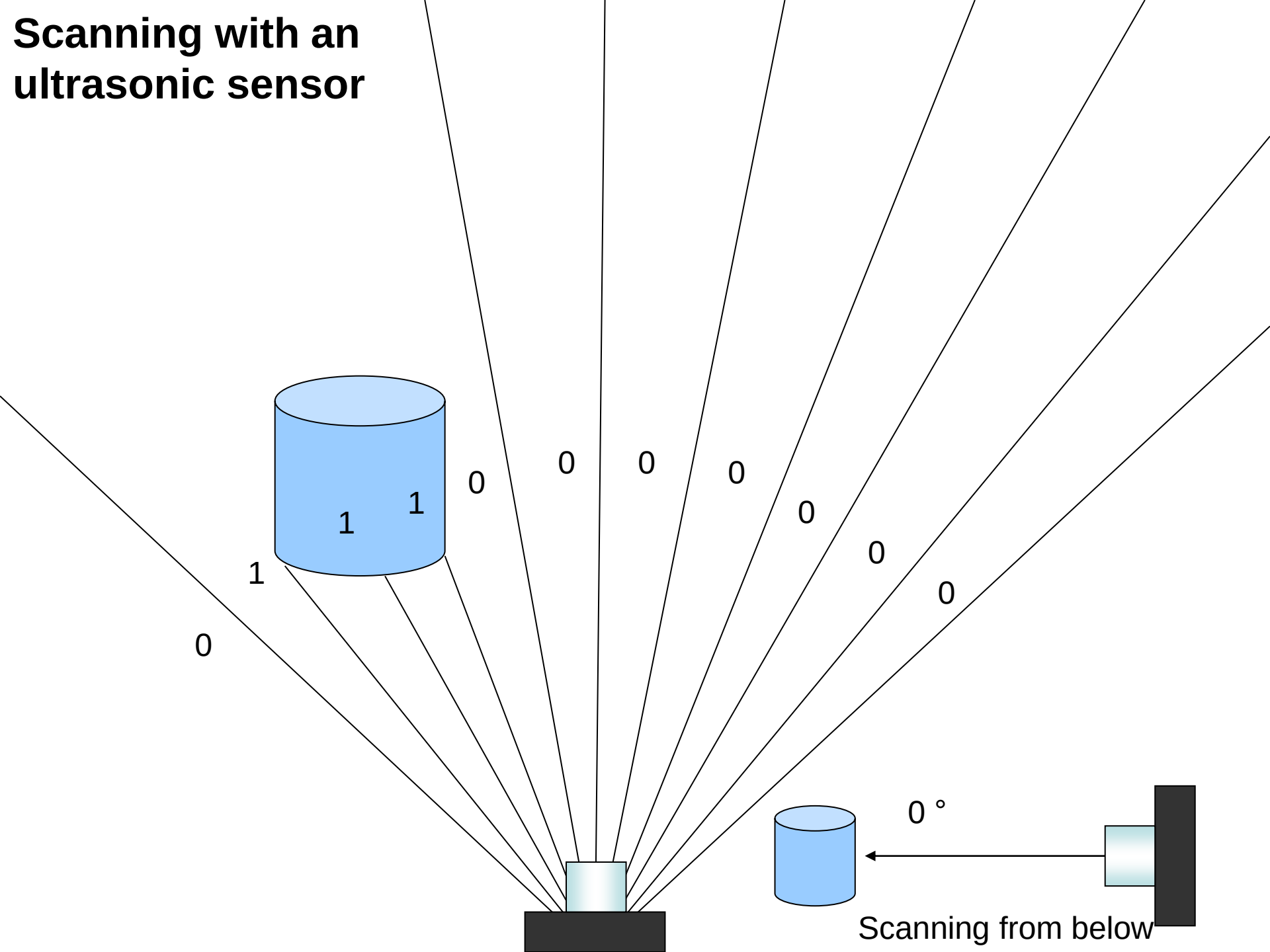
Main Loop Template

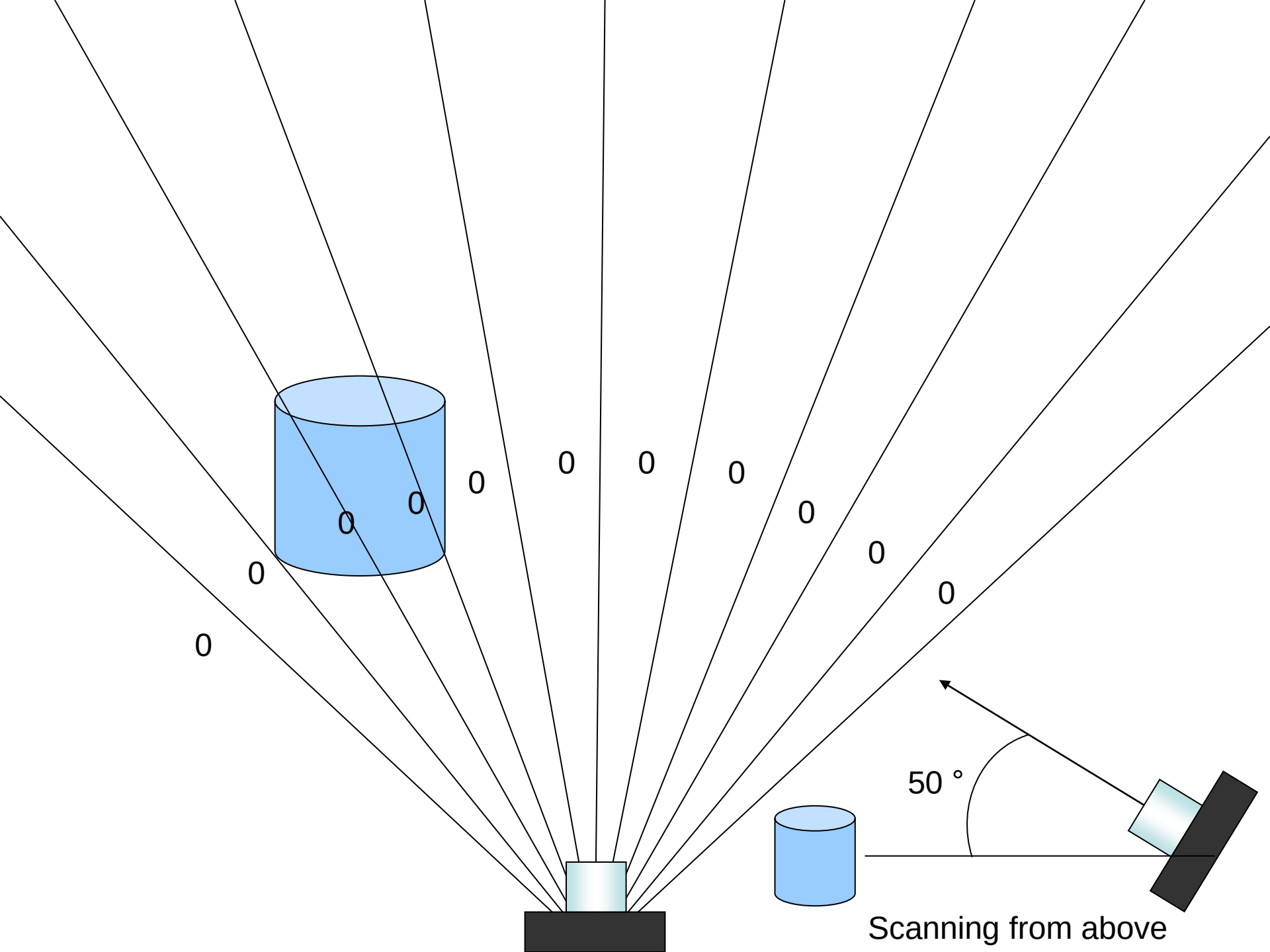


Main loop example (level 2-3):

```
while True:
    H,V,D=scan3D() # scan
    p=model.predict(np.array(D).reshape(1,-1))[0] # class
    if p: # identified ?
        F(5) # push !
    else:
        # move randomly
        direction=random.choice([F,LF,RF])
        direction(random.random())
```

Scanning with an ultrasonic sensor




0
0
0
0
0
0
0
0
0
0
0
50 °
Scanning from above

We get the scan results (“image"):

[0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

Scanning from below

Scanning from above

But how do you teach the machine to identify an object 10 cm wide and 10 cm high?

Some data for machine learning (classification problem)

Features (a set of «images»)

```
x=np.array([
[1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
...
...
...
[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,  1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]
])
```

Classes

```
y=np.array([
0,
1,
1,
1,
...
...
...
0,
0
])
```

if class 1, then the desired object is in front of the robot

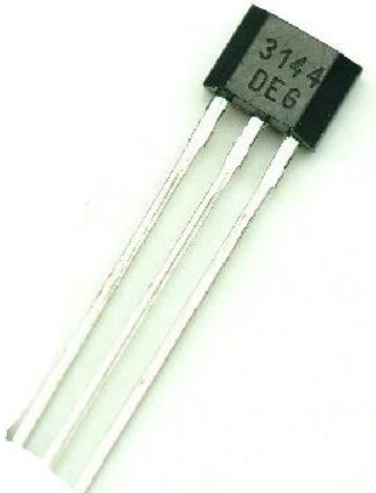
Machine Learning with Decision Tree Gradient Boosting Algorithm

```
# using scikit-learn
from sklearn.ensemble import GradientBoostingClassifier

# create the model
model=GradientBoostingClassifier(n_estimators=10,
                                 learning_rate=0.1,
                                 max_depth=3)

# perform training, but first, using cross-validation, we should make sure that the model has a high score
model.fit(x, y)
```

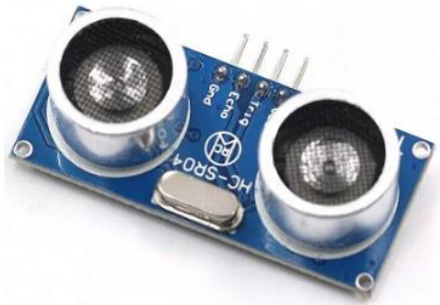
Sensors that can be used

Infrared Obstacle Sensor

each < 2$



Hall sensor + magnet

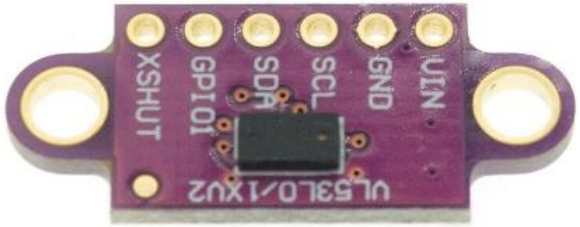


Ultrasonic Distance Sensor

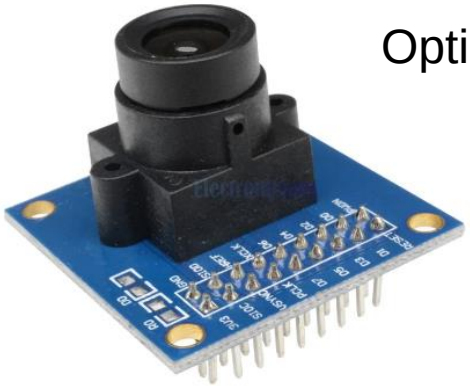
Photoresistor + LED

Optical encoder

3-axis compass + accelerometer



Laser distance sensor VL53L0X



Camera

github.com/vkopey/mechatronics3