Garbage identification

Testing based on the trained model can identify the name of the trained object.

Code path:

```
~/jetcobot_ws/src/jetcobot_garbage_yolov11/Garbage_identification.ipynb
```

The specific code can be viewed in the code path.

Import various libraries

```
import cv2 as cv
import threading
from time import sleep
import ipywidgets as widgets
from IPython.display import display
from single_garbage_identify import single_garbage_identify
model = attempt_load(model_path, map_location=device)
```

The camera shows the garbage identification screen

```
def camera():
    # 打开摄像头 Open camera
    capture = cv.VideoCapture(0)
    capture.set(cv.CAP_PROP_FRAME_WIDTH, 640)
    capture.set(cv.CAP_PROP_FRAME_HEIGHT, 480)

    while capture.isOpened():
        try:
            _, img = capture.read()
            # img = cv.resize(img, (640, 480))
            img = single_garbage.single_garbage_run(img)
            if model == 'Exit':
                cv.destroyAllWindows()
                capture.release()
                break
            imgbox.value = cv.imencode('.jpg', img)[1].tobytes()
        except KeyboardInterrupt:capture.release()
```

Garbage Identification Function

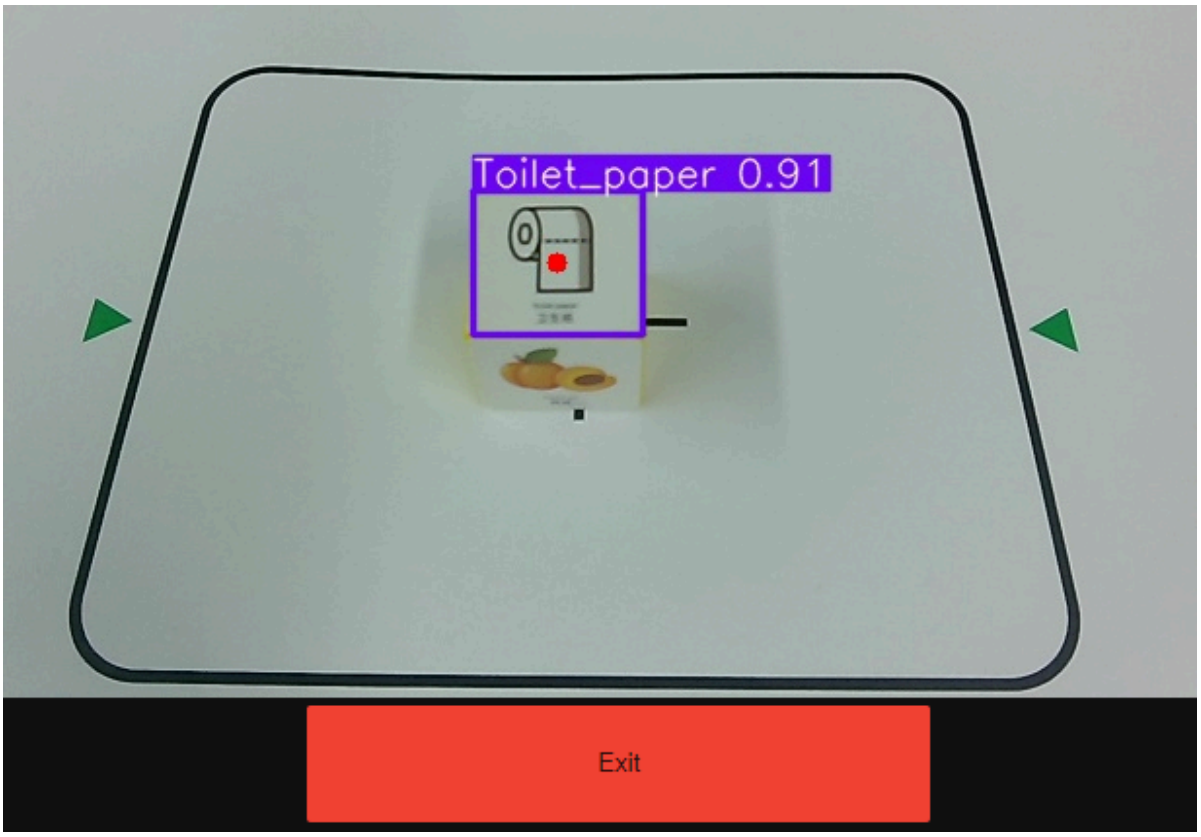
```
def single_garbage_run(self, image):
    self.frame = cv.resize(image, (640, 480))
    try:
    self.garbage_getName()
    except Exception:
    print("sqaure_pos empty")
    return self.frame
```

List of junk names:

```
 def garbage_getName(self):
```

```
        name = "None"
        if self.status == 'waiting':
            self.frame, msg = self.garbage_identify.garbage_run(self.frame)
            for key, pos in msg.items(): name = key
            if name == "Zip_top_can":               (self.garbage_num,
self.garbage_class) = ('00', '01')
            if name == "Old_school_bag":            (self.garbage_num,
self.garbage_class) = ('01', '01')
            if name == "Newspaper":                 (self.garbage_num,
self.garbage_class) = ('02', '01')
            if name == "Book":                      (self.garbage_num,
self.garbage_class) = ('03', '01')
            if name == "Toilet_paper":              (self.garbage_num,
self.garbage_class) = ('04', '02')
            if name == "Peach_pit":                 (self.garbage_num,
self.garbage_class) = ('05', '02')
            if name == "Cigarette_butts":           (self.garbage_num,
self.garbage_class) = ('06', '02')
            if name == "Disposable_chopsticks":     (self.garbage_num,
self.garbage_class) = ('07', '02')
            if name == "Egg_shell":                 (self.garbage_num,
self.garbage_class) = ('08', '03')
            if name == "Apple_core":                (self.garbage_num,
self.garbage_class) = ('09', '03')
            if name == "Watermelon_rind":           (self.garbage_num,
self.garbage_class) = ('10', '03')
            if name == "Fish_bone":                 (self.garbage_num,
self.garbage_class) = ('11', '03')
            if name == "Expired_tablets":           (self.garbage_num,
self.garbage_class) = ('12', '04')
            if name == "Expired_cosmetics":         (self.garbage_num,
self.garbage_class) = ('13', '04')
            if name == "Used_batteries":            (self.garbage_num,
self.garbage_class) = ('14', '04')
            if name == "Syringe":                   (self.garbage_num,
self.garbage_class) = ('15', '04')
            if name == "None":                      (self.garbage_num,
self.garbage_class) = ('None', 'None')
```

Click the JupyterLab toolbar, run all the program blocks, and then pull to the bottom of the program to view the camera screen. Place the building block facing the camera, and the screen will automatically recognize the corresponding garbage name.



Note: The building block needs to face the camera position so that the camera screen can see the picture and name directly as shown in the figure above. If the direction difference is too large, it will affect the recognition rate.