UR3 Robotkar, futószalag és ArUco Marker Felismerő Program Dokumentáció

Sapientia

2023

Készítette:

Antal József

Tartalom

Bevezető

A dokumentáció áttekintést nyújt az alkalmazásról, amely egy UR3-as robotkarhoz és egy futószalagon mozgó darabhoz kapcsolódik, alkalmazás lehetővé teszi az ArUco marker felismerését.

A program három fő részből áll: a futószalag vezérléséből, a robotkar vezérléséből és az ArUco marker felismeréséből.

A futószalag program felelős a darabok mozgatásáért a futószalagon. A program időzítőt használva energiaellátást biztosít a motoroknak, és meghatározott idő elteltével leállítja a futószalagot.

A robotkar program vezérli az UR3-as robotkart, és előre meghatározott pozíciókba mozgatja a darabokat. A program a rtde_control, rtde_receive és rtde_io modulokat használja a robotkar vezérléséhez, a pozíciók lekérdezéséhez és a digitális kimenetek beállításához.

Az ArUco Marker Felismerő program lehetővé teszi az ArUco markerek felismerését a darabokon. A program a cv2 és aruco modulokat használja a kamera képének beolvasásához, a markerek felismeréséhez és a kép megjelenítéséhez.

Robotkar Program

A robot.py fájl tartalmazza az UR3-as robotkarhoz kapcsolódó kódot. A program az rtde_control, rtde_receive és rtde_io modulokat használja a robotkar vezérléséhez, a pozíciók lekérdezéséhez és a digitális kimenetek beállításához.

```
import rtde_control
import rtde_receive
import rtde_io
import time
import subprocess

host = "192.168.98.6"
port = 30004

# control
rtde_c = rtde_control.RTDEControlInterface(host)
# getPosition
rtde_r = rtde_receive.RTDEReceiveInterface(host)
# gripper
rtde_io = rtde_io.RTDEIOInterface(host)

if __name__ == "__main__":
    #p[1] += 0.01         #x tengely fix (előre, hátra)
    #p[2] -= 0.01         #y tengely fix (fel, le)
    #p[3] -= 0.1
    #p[4] += 0.1
    #p[5] += 0.1
    #p[6] += 0.1

    velocity = 1
```

```
acceleration = 1
#TARGET = [Base, Shoulder, Elbow, Wrist1, Wrist2, Wrist3]
target1 = [-0.015, -0.935, 0.85, -1.5, 4.75, 1.6]
target3 = [3, -0.935, 1.2, -1.85, 4.75, 1.6]

rtde_io.setStandardDigitalOut(0, False)

rtde_c.moveJ(target1, velocity, acceleration)

subprocess.call(["python", "fp_elinditas.py"])

p = rtde_r.getActualTCPPose()
#print(f"pose {p}")
target2 = p
target2[2] -= 0.1

rtde_c.moveL(target2, velocity, acceleration)
rtde_io.setStandardDigitalOut(0, True)
rtde_c.moveJ(target1, velocity, acceleration)
rtde_c.moveJ(target3, velocity, acceleration)

p = rtde_r.getActualTCPPose()
#print(f"pose {p}")
target4 = p
target4[2] -= 0.06

rtde_c.moveL(target4, velocity, acceleration)
rtde_io.setStandardDigitalOut(0, False)
rtde_c.stopScript()
```

Funkciók

1. Mozgatási Pozíciók

A program definiál néhány előre meghatározott pozíciót, amelyekre a robotkar mozgatható. Ezeket a pozíciókat a target1, target2, target3, target4 változókban tároljuk.

2. Mozgatás és Kommunikáció

A robotkar programban a következő műveletek történnek:

- A robotkar inicializálása és csatlakozás a megadott IP-címhez és portszámhoz.
- Előre meghatározott pozíciókba mozgatás a moveJ és moveL függvények segítségével.
- Digitális kimenetek beállítása a setStandardDigitalOut függvénnyel.

Használat

A robotkar programot a következőképpen lehet futtatni: python robot.py

Futószalag Program

A fp_elinditas.py fájl tartalmazza a futószalag működését irányító kódot. Ez a program felelős a darabok mozgatásáért a futószalagon.

```
import subprocess
import time

def ticcmd(*args):
    return subprocess.check_output(['ticcmd'] + list(args))

# TIC motorvezérlő inicializálása
ticcmd('--exit-safe-start')

# Kezdő időpont
start_time = time.time()

while True:
    # Motor bekapcsolása
    ticcmd('--energize')

    # Idő ellenőrzése
    current_time = time.time()
    elapsed_time = current_time - start_time

    # Ellenőrzés, hogy eltelt-e már 15 másodperc
    if elapsed_time >= 17:
        break
```

Funkciók

ticcmd(*args)

Ez a függvény a ticcmd parancsot futtatja a motorvezérlőn. A paraméterként kapott argumentumokat átadja a ticcmd parancsnak és visszatér a kimenettel.

Fő Ciklus

A futószalag program fő része egy végtelen ciklus, amelyben a motorok be vannak kapcsolva, és az idő ellenőrzése után leállítja a motorokat.

start_time inicializálása

Ebben a részben az indulási időpontot rögzítjük a program kezdete előtt.

Használat

A futószalag programot a következőképpen lehet futtatni: python fp_elinditas,py

ArUco Marker Felismerő Program

Az aruco_marker_felismero.py fájl tartalmazza az ArUco marker felismerését végző kódot. A program használja a cv2 és aruco modulokat a kamera képének beolvasásához, a markerek felismeréséhez és a kép megjelenítéséhez. Az ArUco marker felismerő program segítségével meghatározható a darabok pontos helye és koordinátái.

```
import cv2
from cv2 import aruco
import matplotlib.pyplot as plt

cap = cv2.VideoCapture(0)

aruco_dict = aruco.Dictionary_get(aruco.DICT_6X6_250)

while True:
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    parameters = aruco.DetectorParameters_create()
    corners, ids, rejectedImgPoints = aruco.detectMarkers(gray, aruco_dict, parameters=parameters)
    frame_markers = aruco.drawDetectedMarkers(frame, corners, ids)
    cv2.imshow('ArUco Detection', frame_markers)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()
```

Használat

A használatához szükséges könyvtárak:

- OpenCV
- rtde

pip install opencv-python rtde

Az ArUco Marker Felismerő programot a következőképpen lehet futtatni:

python aruco_camera,py