Osnove robotike

LV5

Mobilni robotski manipulator

Denis Lazor

Zadatak: Koristeći Lua programski jezik i V-Rep (Coppelia) okruženje isprogramirati sekvencijalno upravljanje mobilnim robotom. Robot treba moći detektirati i skupiti 3 crvene kocke te ih postaviti na platformu na različita mjesta.

NAPOMENA!

➔ Bilo je situacija da V-Rep prebriše modificiranu skriptu pa u prilogu između ostalog šaljem i kod u .txt formatu za svaki slučaj.

➔ **Inicijalizacija**

```
-- ovdje dodajte vasu inicijalizaciju // Added by Denis Lazor

        f = 64/(math.tan(20*math.pi/180))  // Intrizični parametri kamere
        uc = 128                           // -||-
        vc = 128                           // -||-
        state = 0                          // Početno stanje 0
        counter = 1                        // Brojač podignutih kocaka

  -- kraj inicijalizacije
```

➔ Sekvencijalno upravljanje

```
-- sekvencijalno upravljanje -- // Added by Denis Lazor

    -- Getting information from camera --
        result,t0,t1=sim.readVisionSensor(camera)
    blobCount=t1[1]
    if(blobCount>0)then
                xpos=t1[5]
                ypos=t1[6]
                kw=t1[7]
                kh=t1[8]
                sirina=kw*128
                visina=kh*128
                u=uc*xpos
                v=vc*(1-(ypos-kh/2))
                z=((f*0.03)/(v-64))
                z1=z+0.3+0.03
        end


    -- Step 1 -- // Rotates till find red square and centers it on camera
        if (state==0) then

                if (blobCount==0) then
                        rotVel=-1.5
          else
                        rotVel=-0.7

                        if (math.abs(t1[5]-0.50) < 0.0005)then
                                state=1
                                rotVel=0.0
                        end
                end
        end
```

```
-- Step 2 -- // Driving till reach certain distance form square then stops
        if (state == 1) then

                if z1>0.48 then
                        forwBackVel=3
                else
                        state=2
                        forwBackVel=0
                end
        end


-- Step 3 -- // Centering manipulator depending on square position
        if (state == 2) then

                if (t1[5]<0.495) then
                        rotVel=-0.3
                else

                        if (t1[5]>0.505) then
                                rotVel=0.3
                        end

                                if (t1[5]==0.5)then
                                        state=3
                                        rotVel=0.0
                                end
                        end
    end


-- Step 4 -- // Positioning robot manipulator gripper above square using direct kinematics
        if (state == 3) then
        desiredJ={0,-30.91*math.pi/180,-52.42*math.pi/180,-72.68*math.pi/180,-90*math.pi/180}
        aFinished=true
                        for i=1,5,1 do

                                if (math.abs(desiredJ[i]-currentJ[i])>0.002) then
                                        aFinished=false
                                end
                        end

                                if aFinished then
                                        state=4
                                end
        end
```

```
-- Step 5 -- // Positioning robot manipulator gripper on square using inverse kinematics
        if (state == 4) then
                desiredPos = {z1, 0, 0.040}
                ikMode=true
                cFinished=true

                        for i=1,3,1 do

                                if (math.abs(desiredPos[i]-currentPos[i])>0.02) then
                                        cFinished=false
                                end
                        end

                                if cFinished then
                                        state=5
                                StartStateTime = simGetSystemTimeInMilliseconds()
                                end
        end


-- Step 6 -- // Closing gripper
        if (state == 5) then
                bGripper=true;
                OpenGripper=0;
                time = simGetSystemTimeInMilliseconds()

                if(time - StartStateTime > 3000) then
                        state=6
                end
        end


-- Step 7 -- // Liting square object
        if (state == 6) then
        desiredPos = {0.5, 0, 0.4}
                ikMode=true
                dFinished=true

                        for i=1,3,1 do

                                if (math.abs(desiredPos[i]-currentPos[i])>0.02) then
                                        dFinished=false
                                end
                        end

                                if dFinished then
                                        ikMode=false
                                                state=7
                                end
        end
```

```
  -- Step 8 -- // Putting square objects on platform on diffrent places using direct kinematics
if (state == 7) then
      if (counter==1) then

     desiredJ={20*math.pi/180,30*math.pi/180,52*math.pi/180,72*math.pi/180,-  90*math.pi/180}
     eFinished=true

     for i=1,5,1 do

               if (math.abs(desiredJ[i]-currentJ[i])>0.002) then
                         eFinished=false
               end
     end

               if eFinished then
                         state=8
               end
     end

     if (counter==2) then

     desiredJ={0*math.pi/180,30*math.pi/180,52*math.pi/180,72*math.pi/180,-90*math.pi/180}
     eFinished=true

     for i=1,5,1 do

        if (math.abs(desiredJ[i]-currentJ[i])>0.002) then
           eFinished=false
        end
     end

     if eFinished then
        state=8
     end
  end

     if (counter==3) then
     desiredJ={-20*math.pi/180,30*math.pi/180,52*math.pi/180,72*math.pi/180,-90*math.pi/180}
     eFinished=true

     for i=1,5,1 do

        if (math.abs(desiredJ[i]-currentJ[i])>0.002) then
           eFinished=false
        end
     end

     if eFinished then
        state=8
     end
  end

  end
```

➔ U 8 koraku je dodan kod koji s obzirom na veličinu counter-a koji predstavlja broj postavljenih kocaka na platformu , mijenja položaj gdje zadnja kocka treba biti stavljena na način da se se prvi od 5 zglobova (onaj koji ritoira cijelog robota zarotira tako da poslaže kocke jednu pokraj druge. Točnije za 20 stupnjeva za svaku novu kocku dok ne dostigne 3 kocke.

```
-- Step 9 -- // Oppening gripper and stoping program if neccessary
        if (state == 8) then
                bGripper=true;
                OpenGripper=1;
                time = simGetSystemTimeInMilliseconds()

                if(time - StartStateTime > 7000) then
                        bGripper=false;

                        if (counter<3) then
                                state=0
                                counter=counter+1
                        else
                                state=9
                        end
                end
        end

-- kraj sekvencijalnog upravljanja --
```

➔ U koraku 9 je bilo problema da se nakon postavljanja kocke na platformu hvatlaljka nije otvorila zbog čega je povišeno vrijeme čekanja na hvataljku da se otovri na 7000.

REZULTATI: