NAAC GRADE A+
Accredited University

CU CHANDIGARH UNIVERSITY

Introduction to Line Following Robot using

CoppeliaSim
from the creators of V-REP

Dr. Prashant Upadhyaya
Senior Member-IEEE, Fellow-IETE-Australia, IETE (M)-India
Associate Professor, ECE (AU-1)
Chandigarh University

Research Profile -Dr. Prashant Upadhyaya - Google Scholar

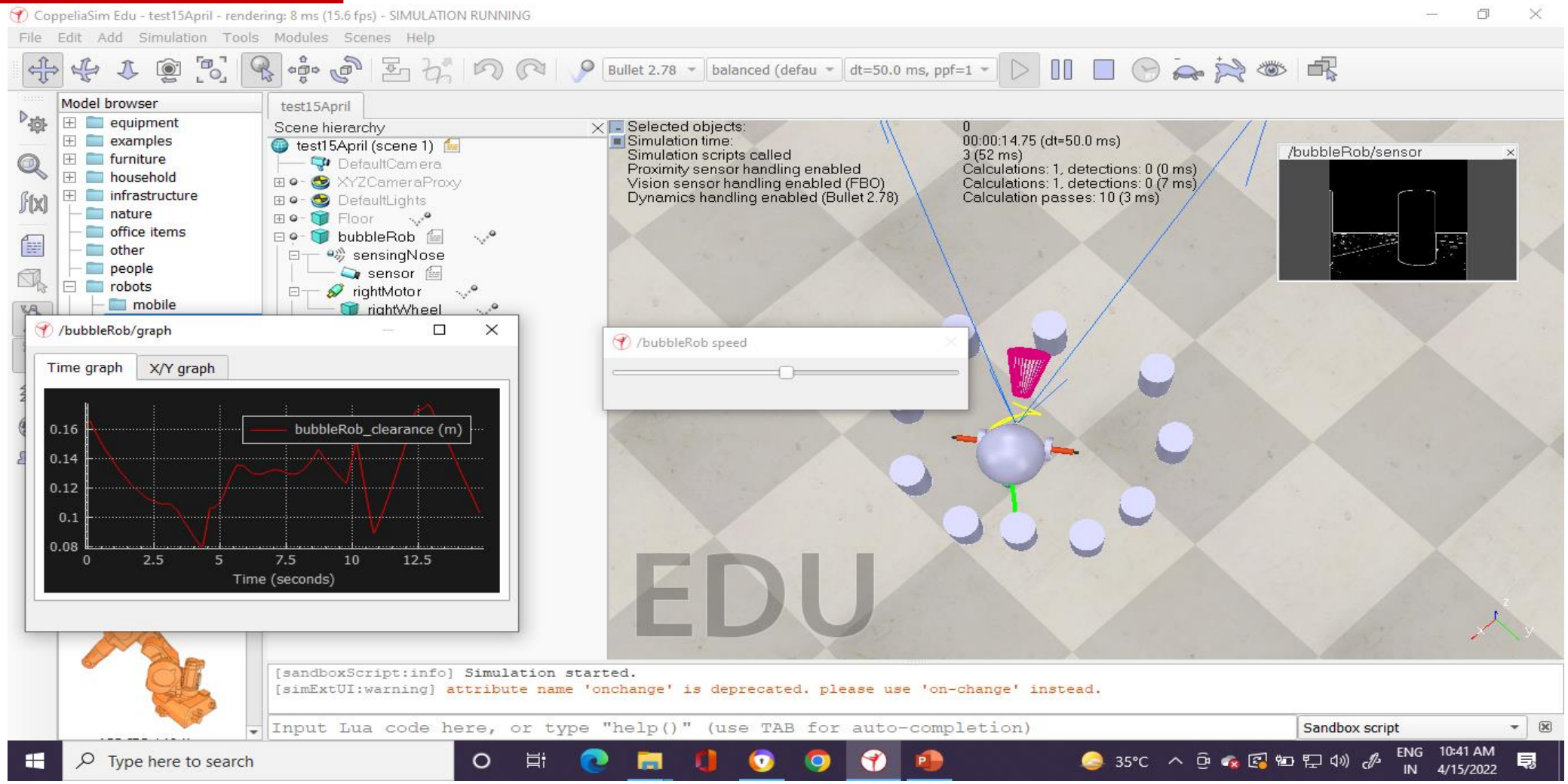
Line Following Robot: Tutorial

In this tutorial we aim at extending the functionality of BubbleRob to let it follow a line on the ground.

Make sure you have fully read and understood the first BubbleRob tutorial.

STEP 1

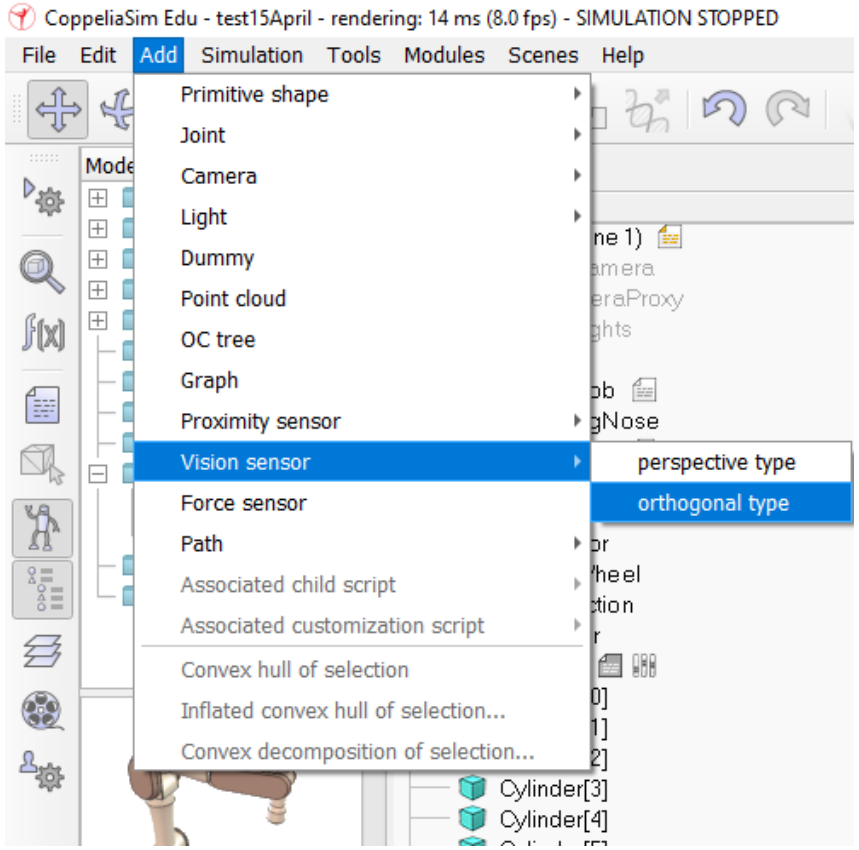
Load the scene of the first BubbleRob tutorial located in scenes/tutorials/BubbleRob.

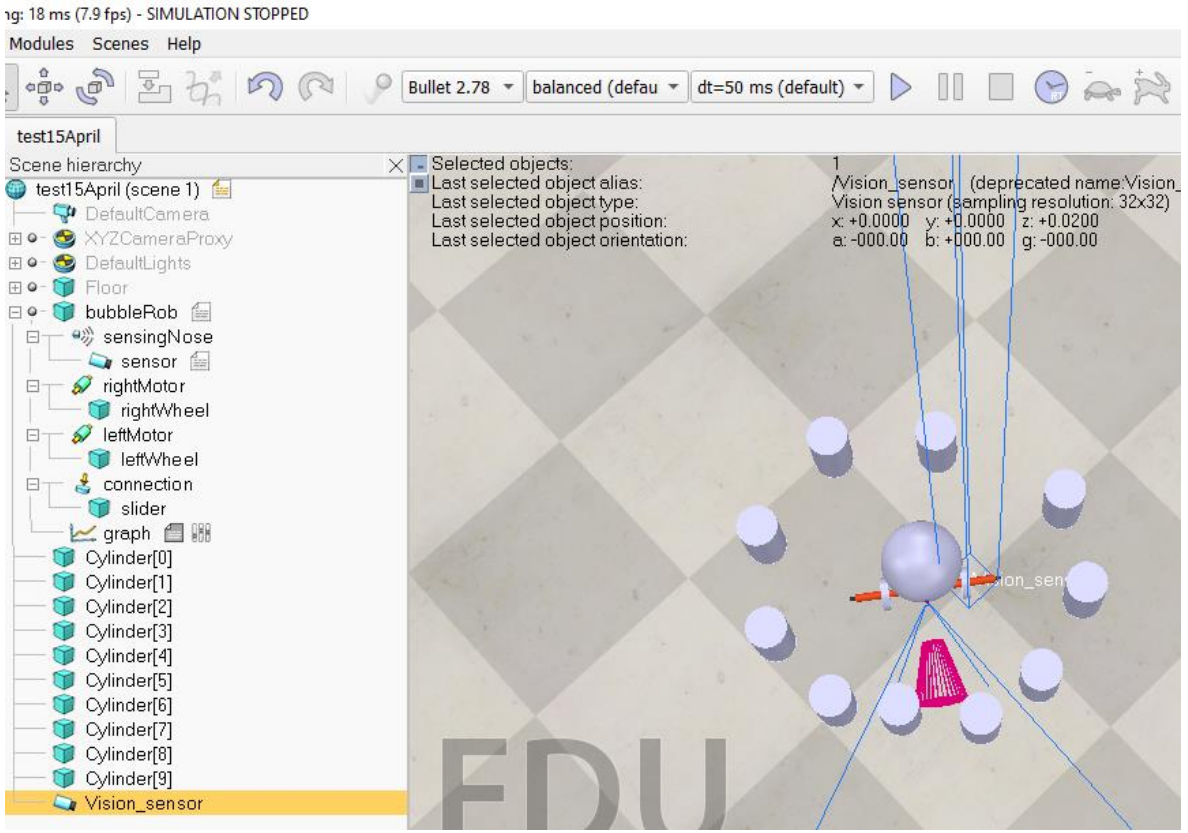


STEP 2

We first create the first of 3 vision sensors that we will attach to the bubbleRob object

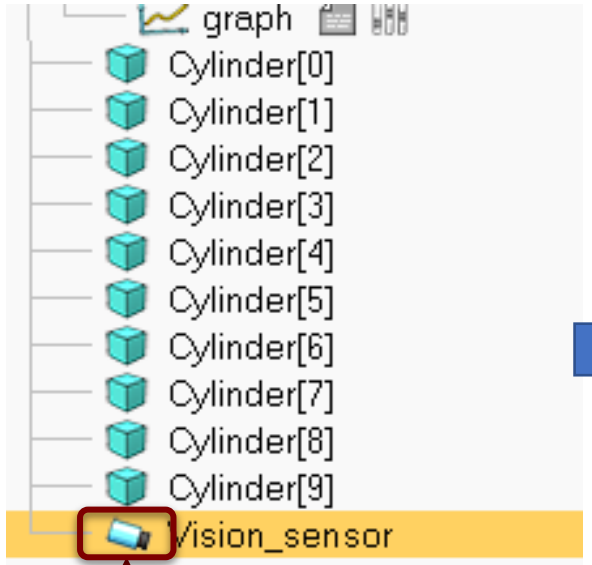
[Menu bar --> Add --> Vision sensor --> Orthographic type]



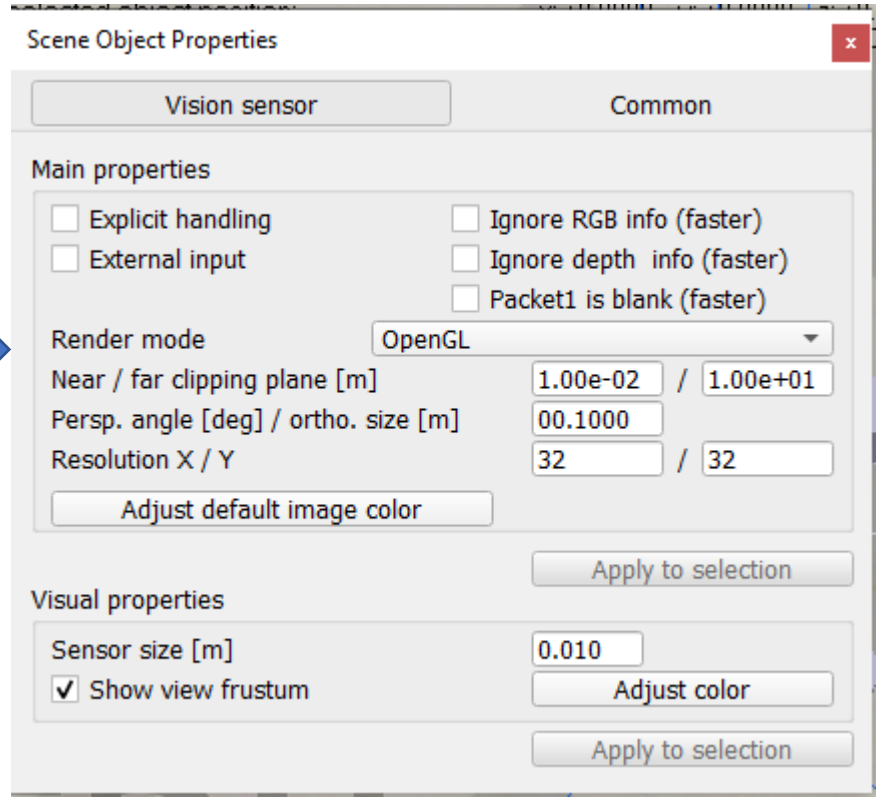


STEP 3

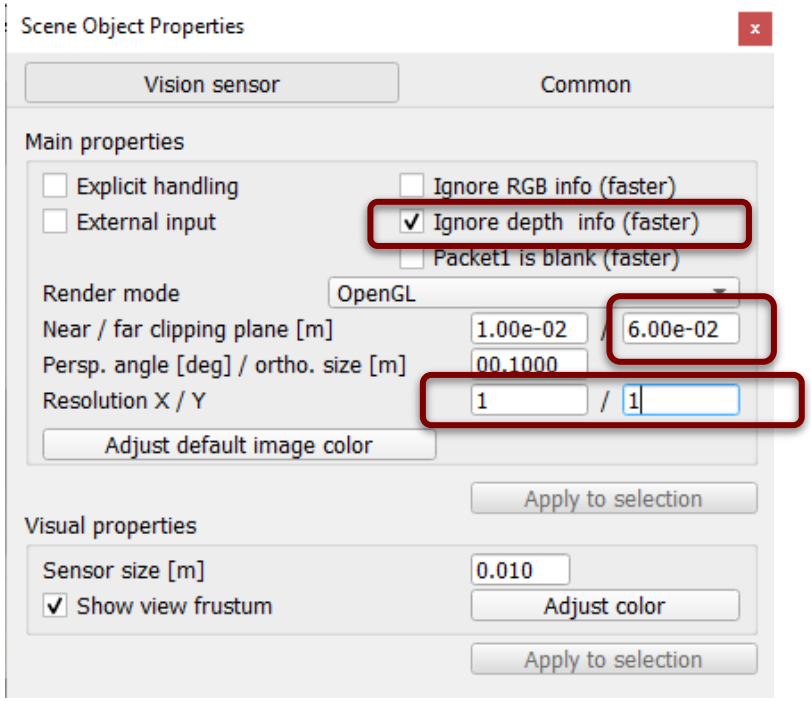
Edit its properties, by double-clicking on the newly created vision sensor icon in the scene hierarchy, and change the parameters to reflect following dialog:



Double Click on camera shape



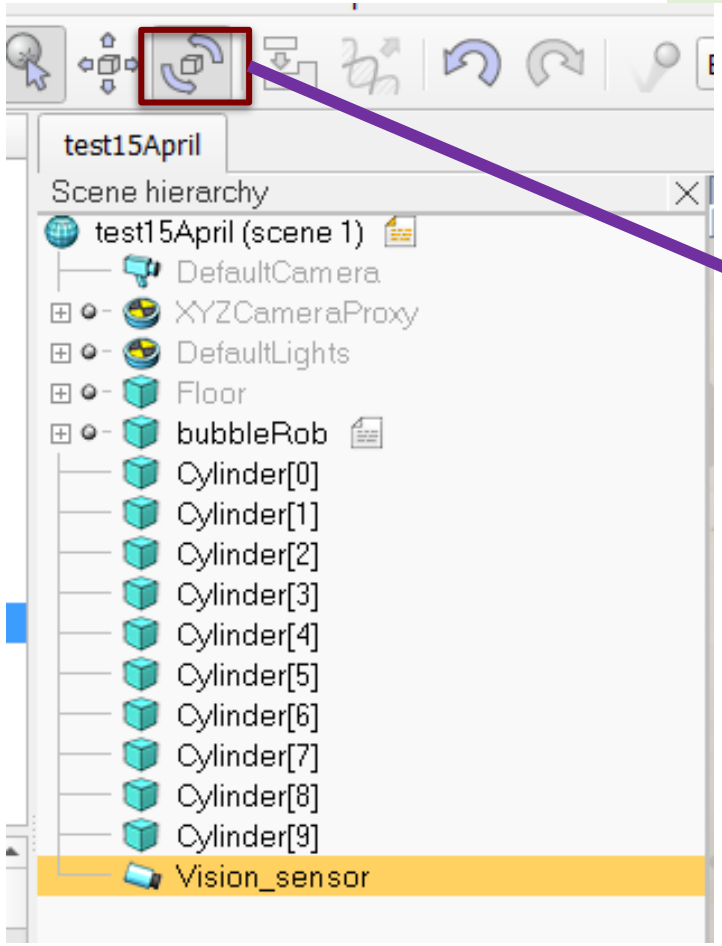
We adjust the far clipping to 6.00e-02 and resolution 1/1

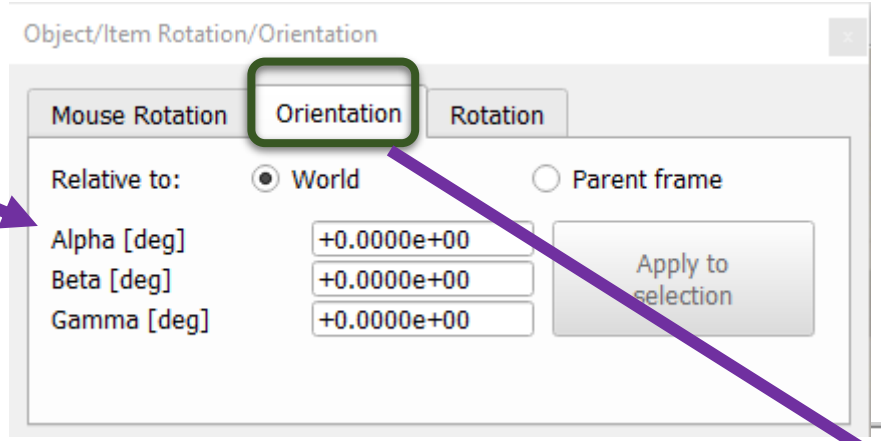


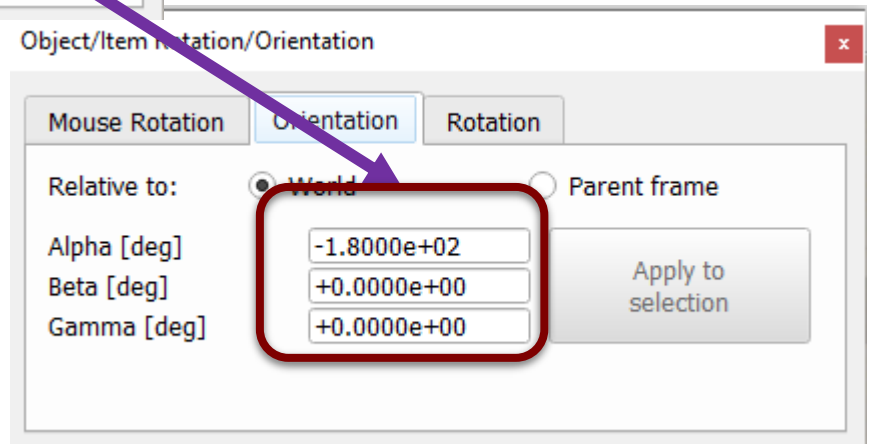
STEP 4

The vision sensor has to be facing the ground, so select it, and in the orientation dialog, on the orientation tab,

Set [180;0;0] for the Alpha-Beta-Gamma items.



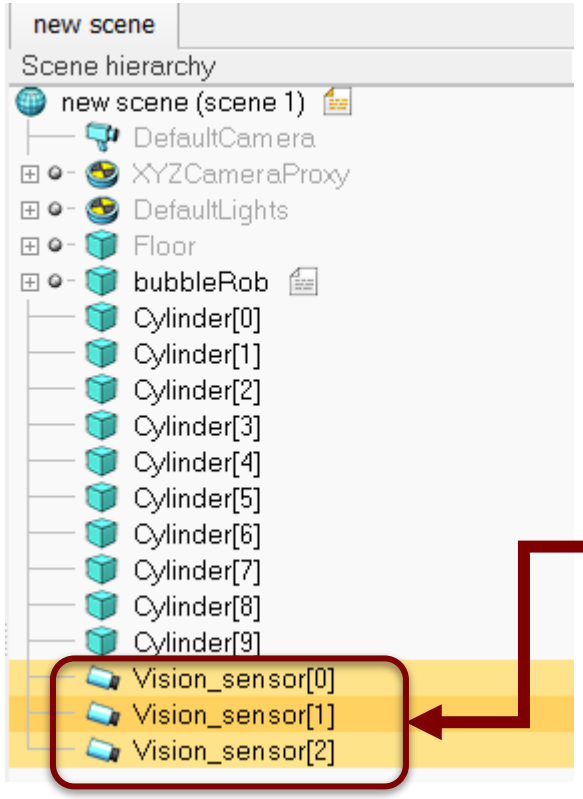




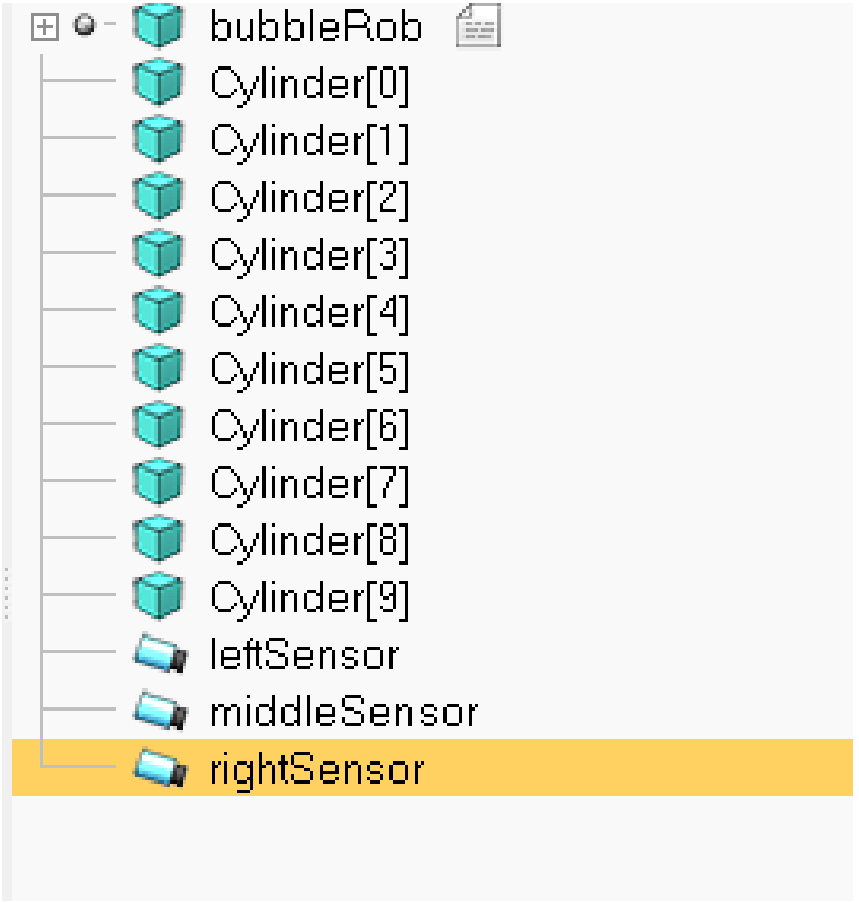
STEP 5

Now copy and paste the vision sensor twice, and adjust its aliases to.

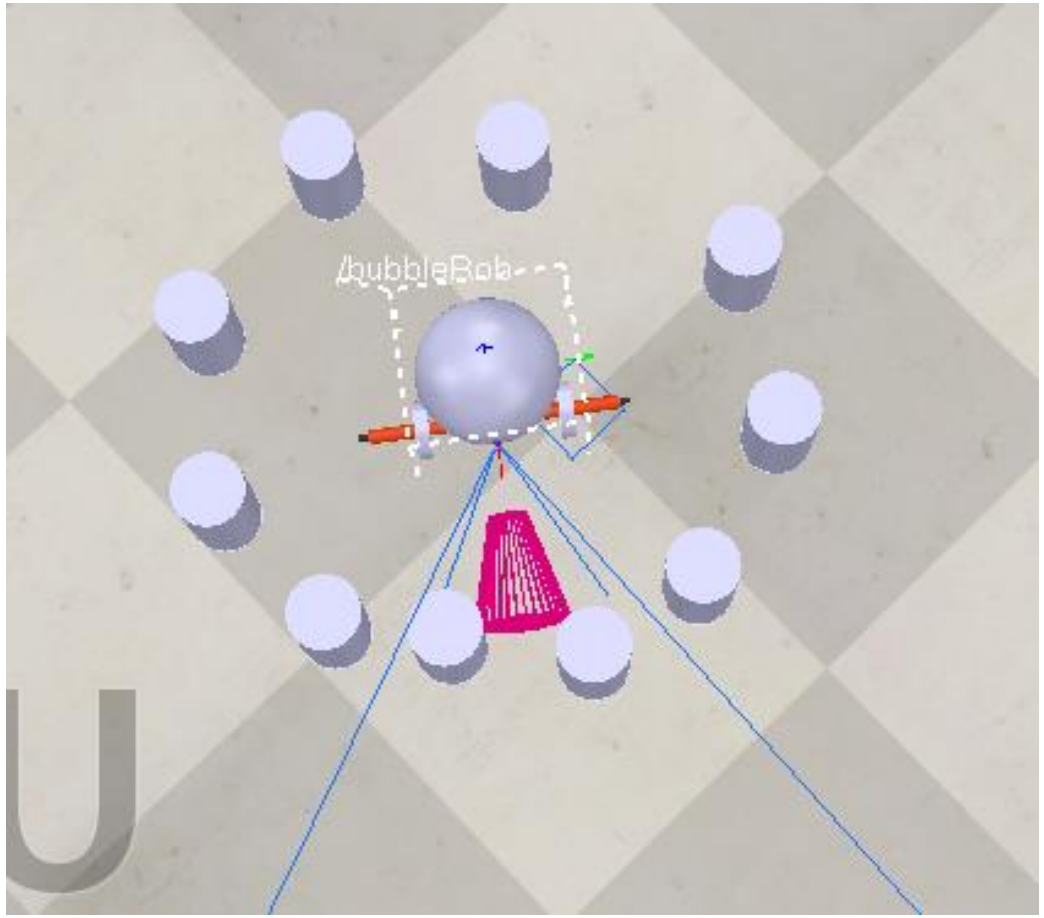
Rename the Vision_Sensor as leftSensor, middleSensor and rightSensor

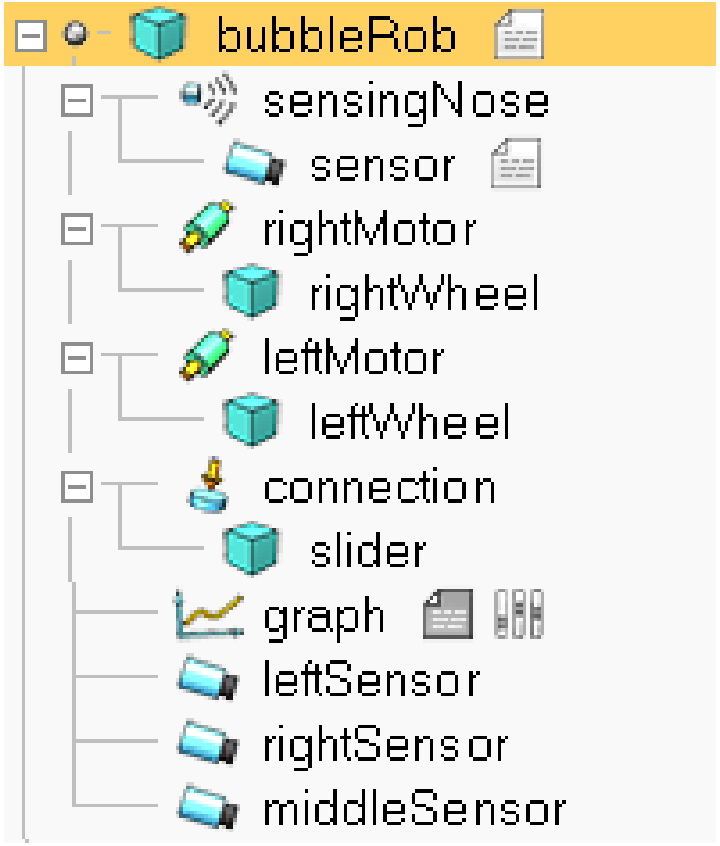


Double Click on this and rename it to leftSensor, middleSensor and rightSensor and Press ENTER



Now let's modify the environment. We can remove a few cylinders in front of BubbleRob. You can see the three sensors. Also make bubbleRob their parent (i.e. attach them to the bubbleRob object).

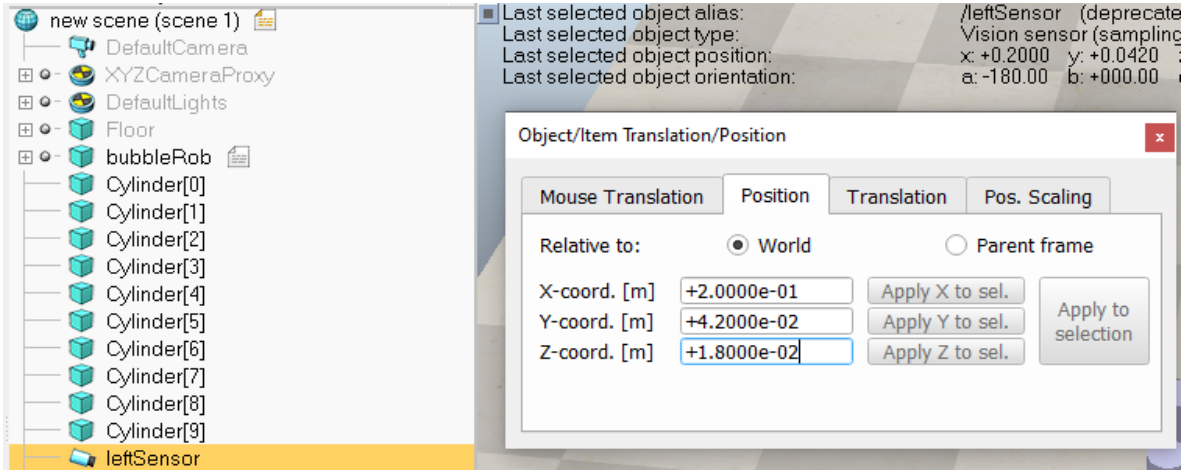


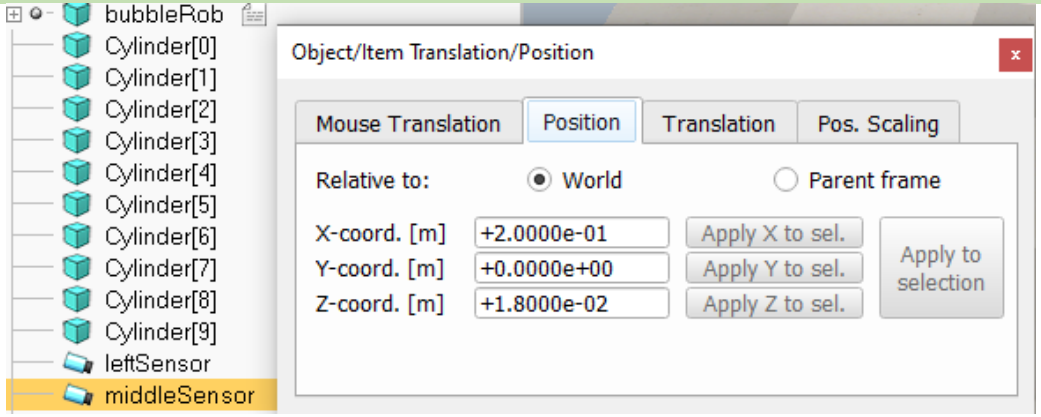


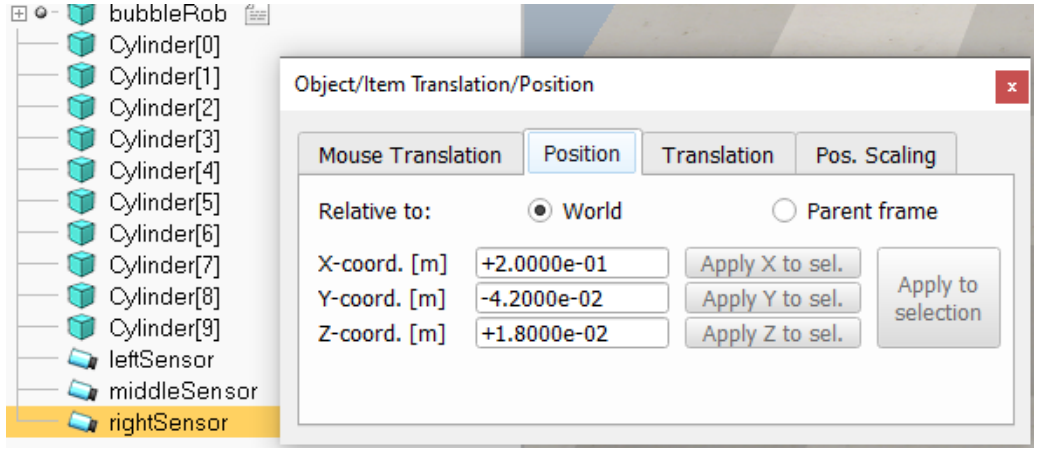
STEP 6

Let's position the sensors correctly.
For that use the position dialog, on the position tab, and set following absolute coordinates:

left sensor: [0.2;0.042;0.018], middle sensor: [0.2;0;0.018], right sensor: [0.2;-0.042;0.018]

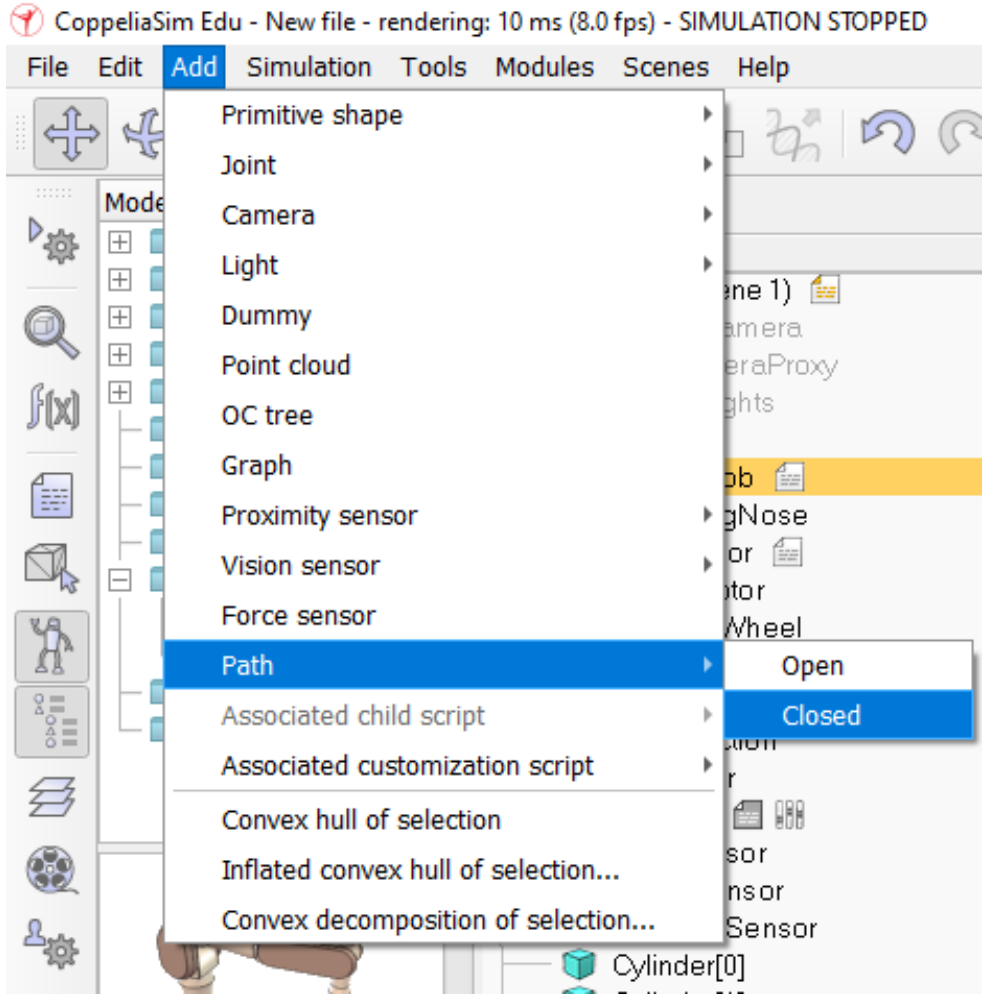


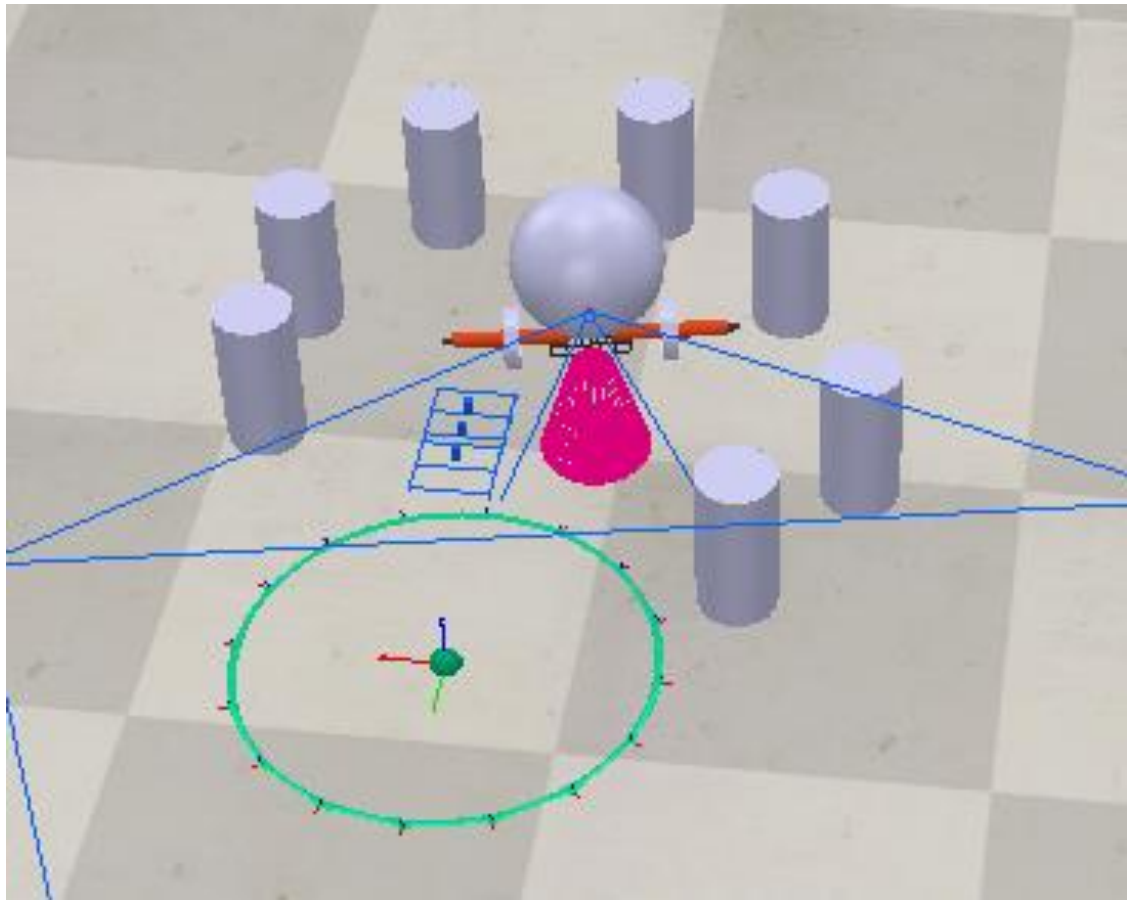




STEP 7

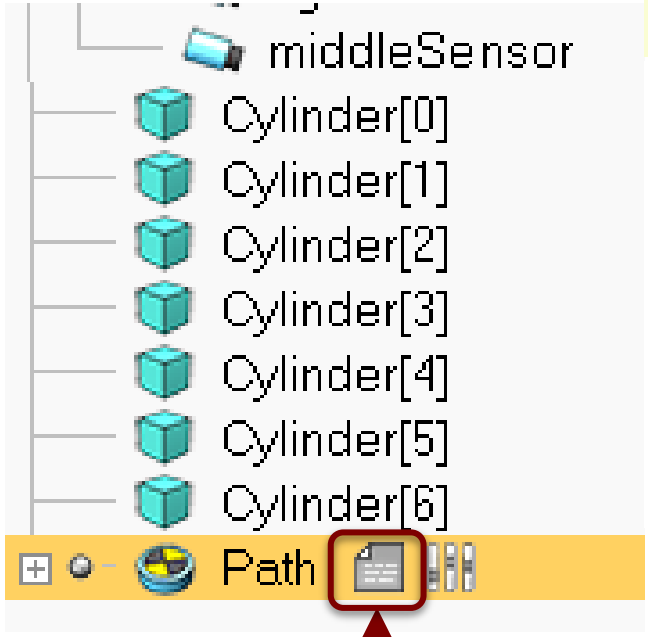
Build the path that the robot will try to follow: click [Menu bar --> Add --> Path --> Closed].





STEP 8

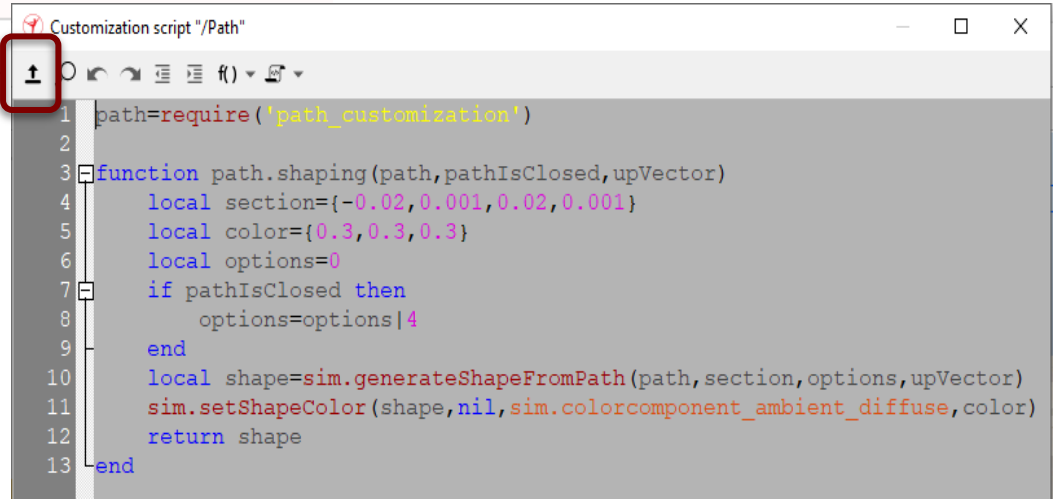
Open the customization script attached to it and replace its content and restart the customization script for the changes to take effect.



Double click on this and copy the code

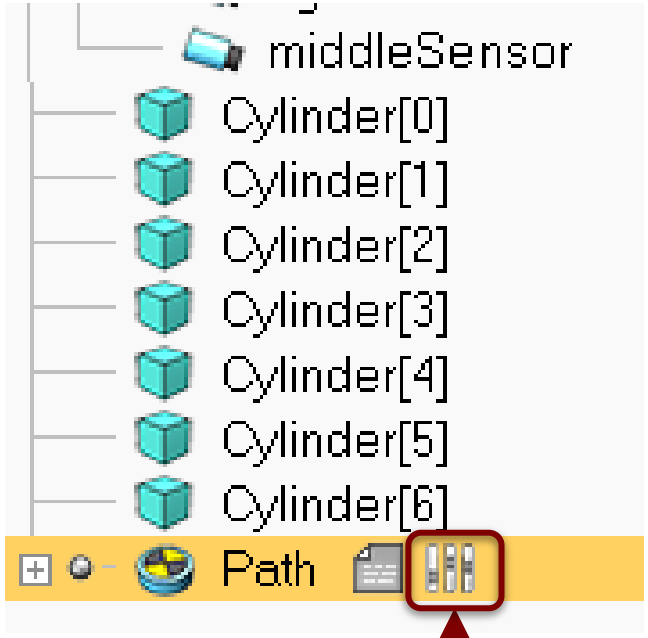
```
path=require('path_customization')

function path.shaping(path,pathIsClosed,upVector)
    local section={-0.02,0.001,0.02,0.001}
    local color={0.3,0.3,0.3}
    local options=0
    if pathIsClosed then
        options=options|4
    end
    local shape=sim.generateShapeFromPath(path,section,options,upVector)
    sim.setShapeColor(shape,nil,sim.colorcomponent_ambient_diffuse,color)
    return shape
end
```

Restart Script

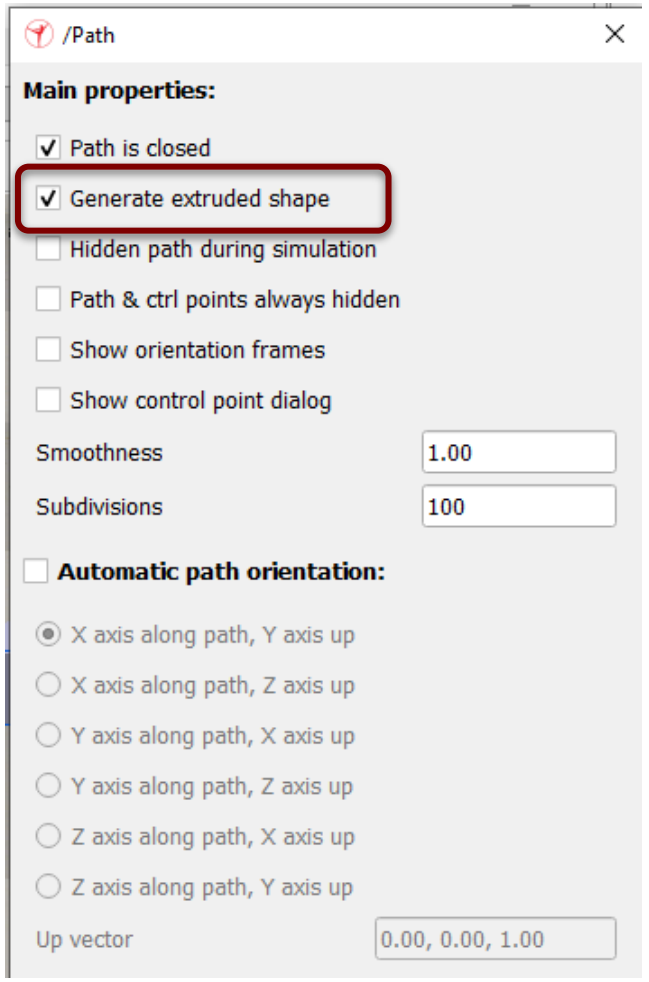


STEP 9

Open the path's user configuration dialog and check the Generate extruded shape checkbox.



Double click on this

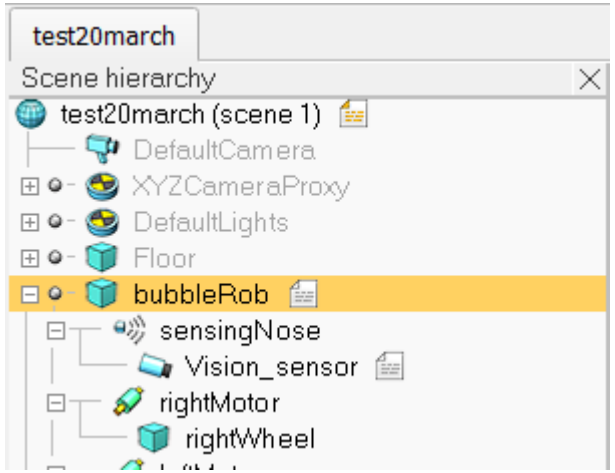


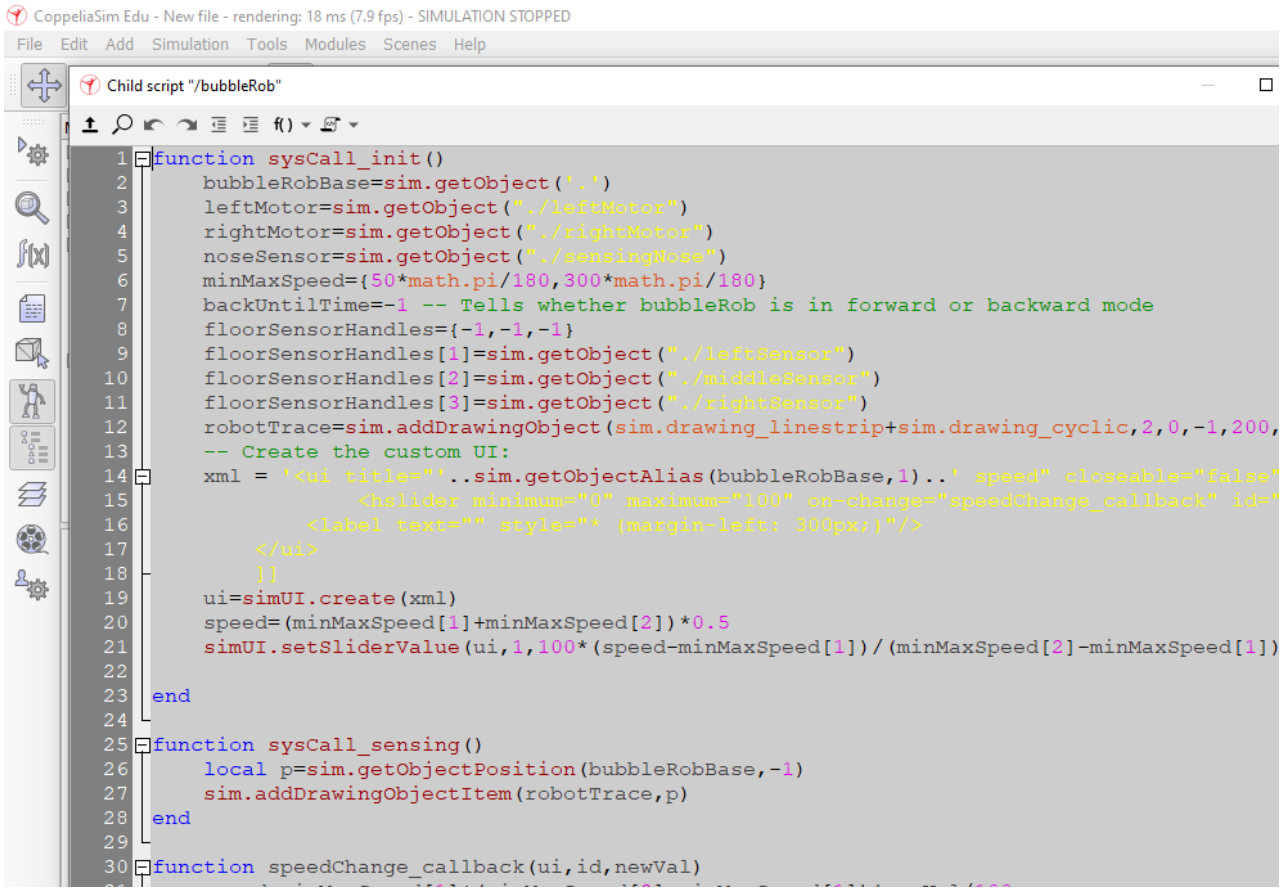
Adding child script:

Control BubbleRob's behavior

STEP 10

Double-click the icon that appeared next to the vision sensor in the scene hierarchy: this opens the child script that we just added. We copy and paste following code into the script editor, then close it:






```
function sysCall_init()
    bubbleRobBase=sim.getObject('.')
    leftMotor=sim.getObject("./leftMotor")
    rightMotor=sim.getObject("./rightMotor")
    noseSensor=sim.getObject("./sensingNose")
    minMaxSpeed={50*math.pi/180,300*math.pi/180}
    backUntilTime=-1 -- Tells whether bubbleRob is in forward or backward mode
    floorSensorHandles={-1,-1,-1}
    floorSensorHandles[1]=sim.getObject("./leftSensor")
    floorSensorHandles[2]=sim.getObject("./middleSensor")
    floorSensorHandles[3]=sim.getObject("./rightSensor")
    robotTrace=sim.addDrawingObject(sim.drawing_linestrip+sim.drawing_cyclic,2,0,-1,200,
    -- Create the custom UI:
    xml = '<ui title="'..sim.getObjectAlias(bubbleRobBase,1)..' speed" closeable="false"
                <hslider minimum="0" maximum="100" on-change="speedChange_callback" id="
            <label text="" style="* {margin-left: 300px;}"/>
        </ui>
        ]]
    ui=simUI.create(xml)
    speed=(minMaxSpeed[1]+minMaxSpeed[2])*0.5
    simUI.setSliderValue(ui,1,100*(speed-minMaxSpeed[1])/(minMaxSpeed[2]-minMaxSpeed[1])

end

function sysCall_sensing()
    local p=sim.getObjectPosition(bubbleRobBase,-1)
    sim.addDrawingObjectItem(robotTrace,p)
end

function speedChange_callback(ui,id,newVal)
```

Child script "/bubbleRob"

```
function speedChange_callback(ui,id,newVal)
    speed=minMaxSpeed[1]+(minMaxSpeed[2]-minMaxSpeed[1])*newVal/100
end

function sysCall_init()
    -- This is executed exactly once, the first time this script is executed
    bubbleRobBase=sim.getObject('.') -- this is bubbleRob's handle
    leftMotor=sim.getObject("./leftMotor") -- Handle of the left motor
    rightMotor=sim.getObject("./rightMotor") -- Handle of the right motor
    noseSensor=sim.getObject("./sensingNose") -- Handle of the proximity sensor
    minMaxSpeed={50*math.pi/180,300*math.pi/180} -- Min and max speeds for each motor
    backUntilTime=-1 -- Tells whether bubbleRob is in forward or backward mode
    robotCollection=sim.createCollection(0)
    sim.addItemToCollection(robotCollection,sim.handle_tree,bubbleRobBase,0)
    distanceSegment=sim.addDrawingObject(sim.drawing_lines,4,0,-1,1,{0,1,0})
    robotTrace=sim.addDrawingObject(sim.drawing_linestrip+sim.drawing_cyclic,2,0,-1,200,{1,1
    graph=sim.getObject('./graph')
    distStream=sim.addGraphStream(graph,'bubbleRob clearance','m',0,{1,0,0})
    -- Create the custom UI:
        xml = '<ui title="'..sim.getObjectAlias(bubbleRobBase,1)..' speed" closeable="false"
        <hslider minimum="0" maximum="100" onchange="speedChange_callback" id="1"/>
        <label text="" style="* {margin-left: 300px;}"/>
        </ui>
        ]]
    ui=simUI.create(xml)
    speed=(minMaxSpeed[1]+minMaxSpeed[2])*0.5
    simUI.setSliderValue(ui,1,100*(speed-minMaxSpeed[1])/(minMaxSpeed[2]-minMaxSpeed[1]))
```

prashant.e9437@cumail.in

Mb: 9411047357

Thank You!