

# Introducción a PyBullet

Paula

Abril 2025

## ¿Qué es PyBullet?

**PyBullet** es un módulo de *Python* para la simulación de física en tiempo real. Está basado en el motor físico **Bullet**, una biblioteca rápida y de código abierto utilizada en videojuegos, robótica y aprendizaje por refuerzo.

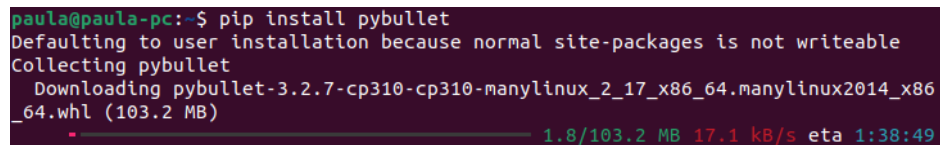
## Características Principales

- Simulación de cuerpos rígidos (gravedad, masas, etc.)
- Detección de colisiones
- Cinemática y dinámica para robots
- Simulación de sensores (cámaras, IMUs, sensores de fuerza)
- API sencilla en Python

## Instalación

Instalar con pip:

```
pip install pybullet
Sudo apt-get install python3
sudo apt install python3-pip
sudo pip3 install pybullet
python3 -m pip show pybullet
```



```
paula@paula-pc:~$ pip install pybullet
Defaulting to user installation because normal site-packages is not writeable
Collecting pybullet
  Downloading pybullet-3.2.7-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (103.2 MB)
    1.8/103.2 MB 17.1 kB/s eta 1:38:49
```

Figura 1: Instalación sugerida

```
paula@paula-pc: ~  
pip._vendor.urllib3.exceptions.ReadTimeoutError: HTTPSConnectionPool(host='files.pythonhosted.org', port=443): Read timed out.  
paula@paula-pc:~$ pip install pybullet  
Defaulting to user installation because normal site-packages is not writeable  
Collecting pybullet  
  Downloading pybullet-3.2.7-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (103.2 MB)  
    4.9/103.2 MB 79.6 kB/s eta 0:20:35  
^Z  
[1]+  Stopped                  pip install pybullet  
paula@paula-pc:~$ sudo apt-get install python3  
[sudo] password for paula:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
python3 is already the newest version (3.10.6-1~22.04.1).  
python3 set to manually installed.  
The following packages were automatically installed and are no longer required:  
  linux-headers-6.2.0-26-generic linux-hwe-6.2-headers-6.2.0-26  
  linux-image-6.2.0-26-generic linux-modules-6.2.0-26-generic  
  linux-modules-extra-6.2.0-26-generic  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
paula@paula-pc:~$
```

Figura 2: Instalación de python3

```
paula@paula-pc:~$ sudo apt-get install python3  
[sudo] password for paula:  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
python3 is already the newest version (3.10.6-1~22.04.1).  
python3 set to manually installed.  
The following packages were automatically installed and are no longer required:  
  linux-headers-6.2.0-26-generic linux-hwe-6.2-headers-6.2.0-26  
  linux-image-6.2.0-26-generic linux-modules-6.2.0-26-generic  
  linux-modules-extra-6.2.0-26-generic  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
paula@paula-pc:~$ sudo apt install python3-pip  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
python3-pip is already the newest version (22.0.2+dfsg-1ubuntu0.5).  
The following packages were automatically installed and are no longer required:  
  linux-headers-6.2.0-26-generic linux-hwe-6.2-headers-6.2.0-26  
  linux-image-6.2.0-26-generic linux-modules-6.2.0-26-generic  
  linux-modules-extra-6.2.0-26-generic  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
paula@paula-pc:~$
```

Figura 3: Instalación pip

```
paula@paula-pc: ~  
python3 is already the newest version (3.10.6-1~22.04.1).  
python3 set to manually installed.  
The following packages were automatically installed and are no longer required:  
  linux-headers-6.2.0-26-generic linux-hwe-6.2-headers-6.2.0-26  
  linux-image-6.2.0-26-generic linux-modules-6.2.0-26-generic  
  linux-modules-extra-6.2.0-26-generic  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
paula@paula-pc:~$ sudo apt install python3-pip  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
python3-pip is already the newest version (22.0.2+dfsg-1ubuntu0.5).  
The following packages were automatically installed and are no longer required:  
  linux-headers-6.2.0-26-generic linux-hwe-6.2-headers-6.2.0-26  
  linux-image-6.2.0-26-generic linux-modules-6.2.0-26-generic  
  linux-modules-extra-6.2.0-26-generic  
Use 'sudo apt autoremove' to remove them.  
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.  
paula@paula-pc:~$ sudo pip3 install pybullet  
Collecting pybullet  
  Downloading pybullet-3.2.7-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (103.2 MB)  
1.2/103.2 MB 102.9 kB/s eta 0:16:31
```

Figura 4: Instalación pybullet

```
paula@paula-pc: ~  
.pythonhosted.org', port=443): Read timed out.  
paula@paula-pc:~$ sudo pip3 install pybullet  
Collecting pybullet  
  Downloading pybullet-3.2.7-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (103.2 MB)  
103.2/103.2 MB 104.6 kB/s eta 0:00:00  
Installing collected packages: pybullet  
Successfully installed pybullet-3.2.7  
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system package manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv  
paula@paula-pc:~$ python3 -m pip show pybullet  
Name: pybullet  
Version: 3.2.7  
Summary: Official Python Interface for the Bullet Physics SDK specialized for Robotics Simulation and Reinforcement Learning  
Home-page: https://github.com/bulletphysics/bullet3  
Author: Erwin Coumans, Yunfei Bai, Jasmine Hsu  
Author-email: erwincoumans@google.com  
License: zlib  
Location: /usr/local/lib/python3.10/dist-packages  
Requires:  
Required-by:  
paula@paula-pc:~$
```

Figura 5: inicialización pybullet

## Utilidades con PyBullet:

- Simular robots como UR5, KUKA, Baxter, PR2, etc.
- Importar y animar modelos URDF (usados en ROS)
- Usar cinemática inversa para mover brazos robóticos
- Recoger datos de sensores
- Crear entornos físicos personalizados
- Entrenar agentes con aprendizaje por refuerzo (RL)

PyBullet Quickstart Guide

## Simulación Brazo robotico en Pybullet

Se realiza la simulación de un brazo robotico con 7 grados de libertad, cuyo diseño es tomado del siguiente repositorio:

*Kuka<sub>e</sub>xperimental*

En este ejemplo de simulación, se busca es un acercamiento inicial con la herramienta de Pybullet para generar movimiento del brazo previamente diseñado. Se realiza en 3 etapas,

iniciando con la presentación de modulo y las funciones necesarias para la carga del mundo y paramentros basicos del robot, paramentrizando variables, como la gravedad y los angulos de giro para cada articulación. Continúa con una adición en codigo, logrando la fijación del punto incial del brazo para que no pierda estabilidad al momento de realizar el movimiento y finalmente, se agrega lineas de código que presentan el debedio manejo matematico de los motores para cada articulación.

```

Codigo inicial para estelbecer parametros del robot y genera la
simulacion de una caida, buscando una torcion de 0.2 metros en
cada articulacion sin una base fija

#importa librerias necesarias
import pybullet
import time
import pybullet_data

Genera Gui de pybullet
physicsClient= pybullet.connect(pybullet.GUI)

Set PybulletData
pybullet.setAdditionalSearchPath(pybullet_data.getDataPath())

load plane - simulation ground from import pybullet_data
planeID=pybullet.loadURDF("plane.urdf")

Load Robot from directory
robot= pybullet.loadURDF("kuka_experimental/kuka_lbr_iiwa_support/urdf
/lbr_iiwa_14_r820.urdf")

How many joints
pybullet.getNumJoints(robot)

Get info position, orientation
position, orientation=pybullet.getBasePositionAndOrientation(robot)

Info for Joint No.7
pybullet.getJointInfo(robot,7)

joint_positions=[j[0] for j in pybullet.getJointStates(robot,range(7))
]
joint_positions

Info for world orientation
world_position, world_orientation=pybullet.getLinkState(robot,2)[:2]

world_position
world_orientation

pybullet.setGravity(0,0,-9.81)

No real time Simulation - by default
pybullet.setRealTimeSimulation(0)

pybullet.setJointMotorControlArray(robot, range(7), pybullet.
POSITION_CONTROL, targetPositions=[0.2]*7)

simula ejecucion dado que no esta en tiempo real
for _ in range(10000):
    pybullet.stepSimulation()
    time.sleep(1./240.)

```

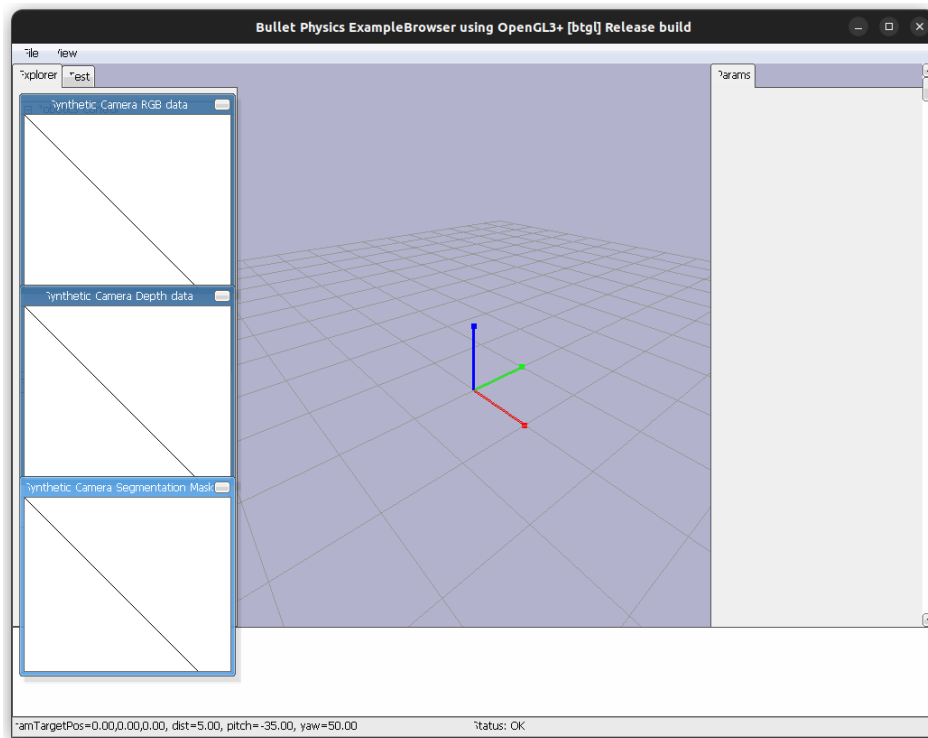


Figura 6: Pybullet

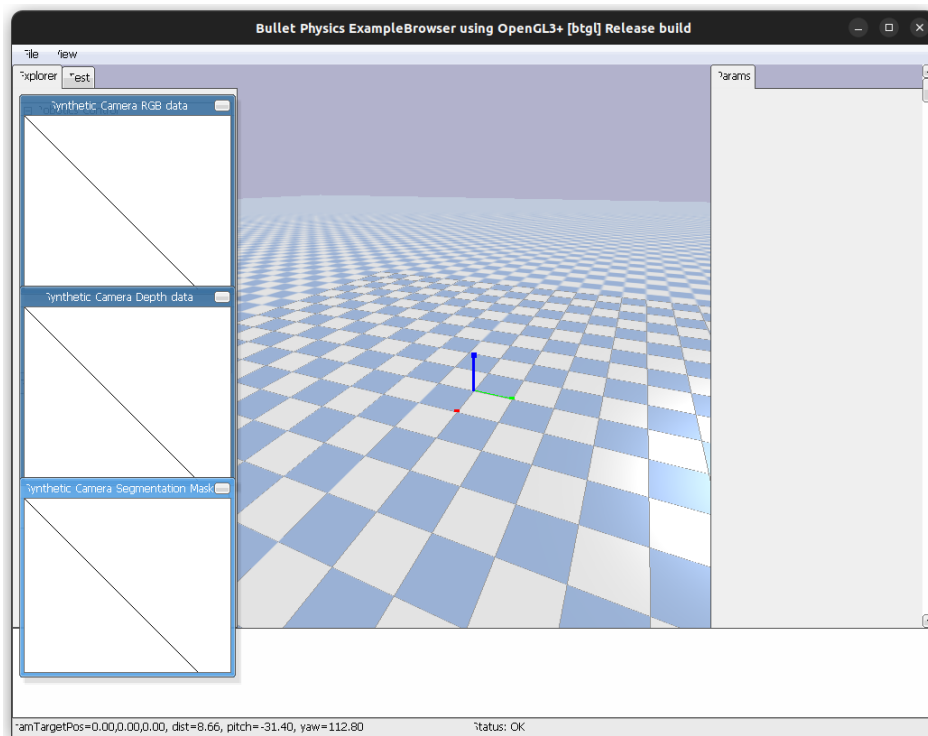


Figura 7: Piso de la simulación

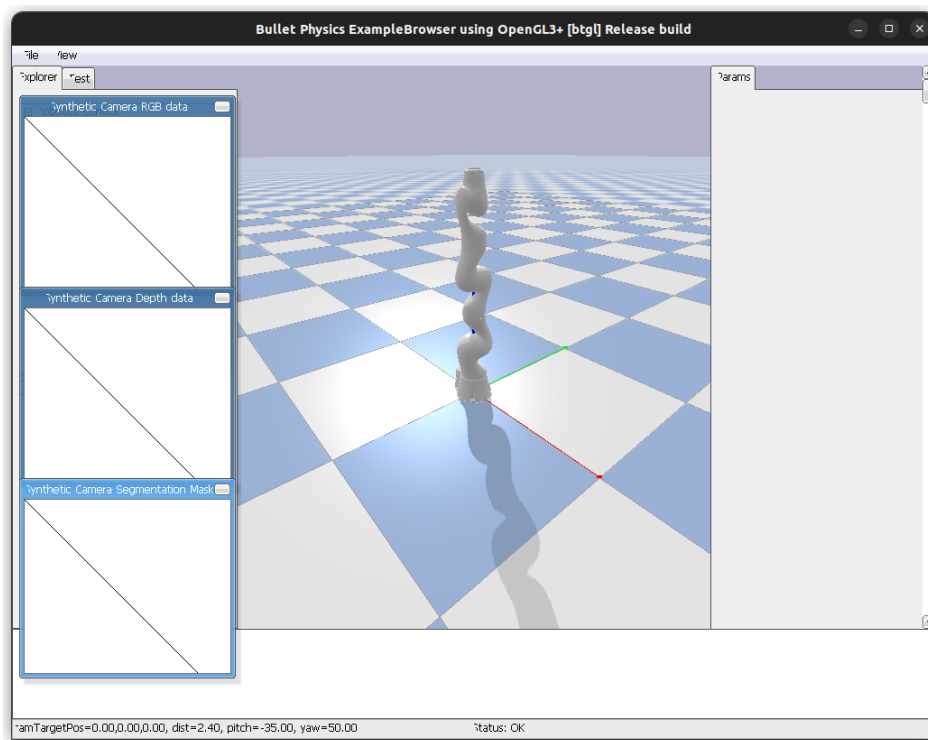


Figura 8: Carga de brazo robotico

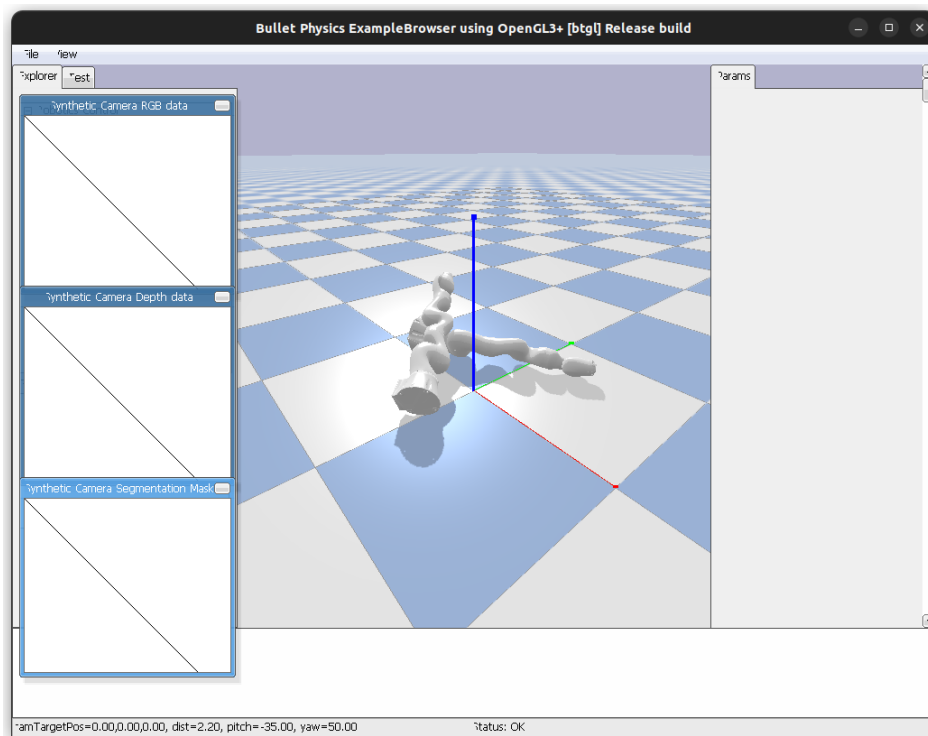


Figura 9: Resultado simulación inicial

## Código simulación 2:

Esta version agrega una base fija permitiendo que se pueda realizar la torcion de 1.5 grados en cada articulacion

```
Reset simulation
pybullet.resetSimulation()

Start Over
import pybullet
import time
import pybullet_data

physicsClient= pybullet.connect(pybullet.GUI)
pybullet.setAdditionalSearchPath(pybullet_data.getDataPath())
planeID=pybullet.load("plane.urdf")
robot= pybullet.loadURDF("kuka_experimental/kuka_lbr_iiwa_support/urdf
    /lbr_iiwa_14_r820.urdf",[0,0,0],useFixedBase=1)
pybullet.setGravity(0,0,-9.81)
pybullet.setRealTimeSimulation(0)
pybullet.setJointMotorControlArray(robot, range(7), pybullet.
    POSITION_CONTROL, targetPositions=[1.5]*7)

for _ in range(300):
    pybullet.stepSimulation()
    time.sleep(1./10.)
```

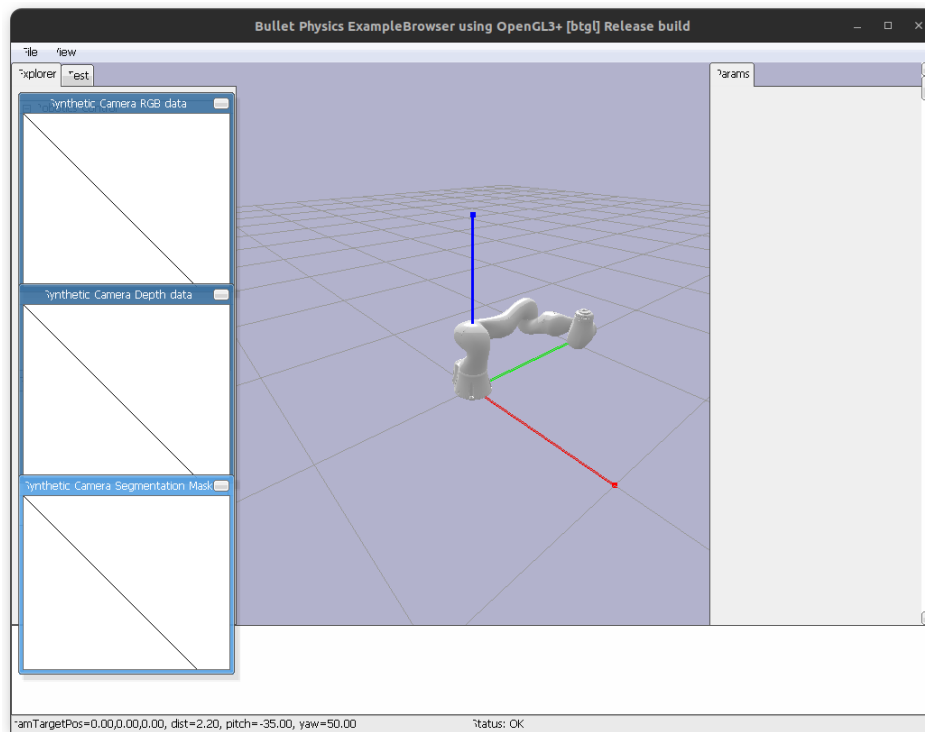


Figura 10: Resultado simulación 2



## Código simulación 2:

```
#Reset simulation
import pybullet
import time
import pybullet_data

physicsClient= pybullet.connect(pybullet.GUI)
pybullet.setAdditionalSearchPath(pybullet_data.getDataPath())
planeID=pybullet.loadURDF("plane.urdf")
robot= pybullet.loadURDF("kuka_experimental/kuka_lbr_iiwa_support/urdf
    /lbr_iiwa_14_r820.urdf",[0,0,0],useFixedBase=1)
pybullet.setGravity(0,0,-9.81)
pybullet.setRealTimeSimulation(0)
Orientation=pybullet.getQuaternionFromEuler([3.14,0.,0.])

targetPositionsJoints=pybullet.calculateInverseKinematics(robot
    ,7,[0.1,0.1,0.4],targetOrientation=Orientation)

pybullet.setJointMotorControlArray(robot, range(7),pybullet.
    POSITION_CONTROL,targetPositions=targetPositionsJoints)

for _ in range(300):
    pybullet.stepSimulation()
    time.sleep(1./10.)
```

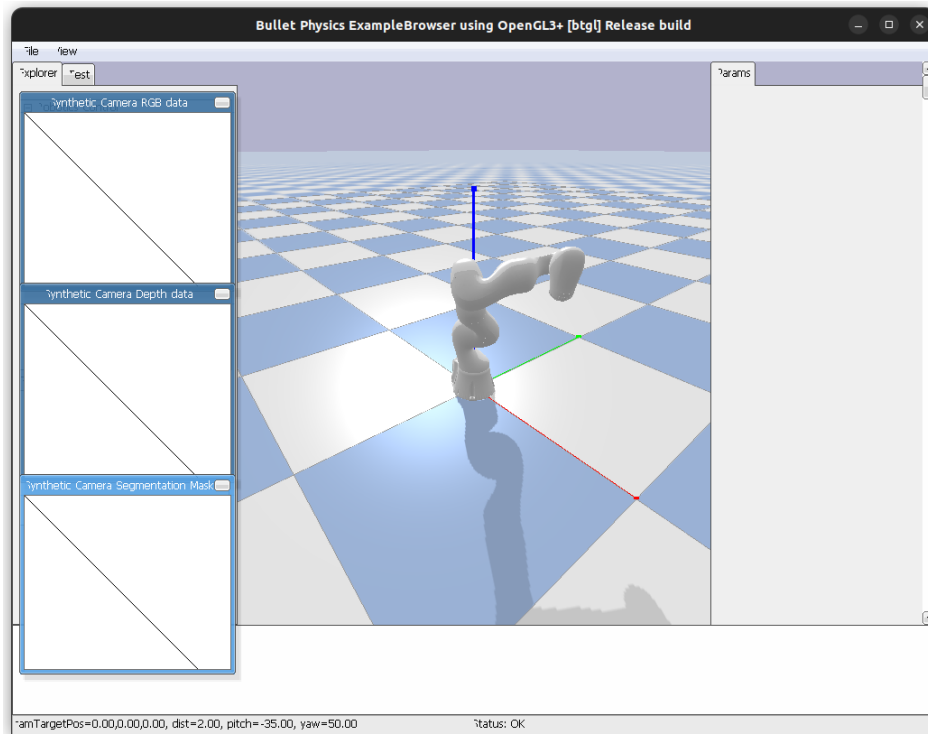


Figura 11: Resultado simulación 3

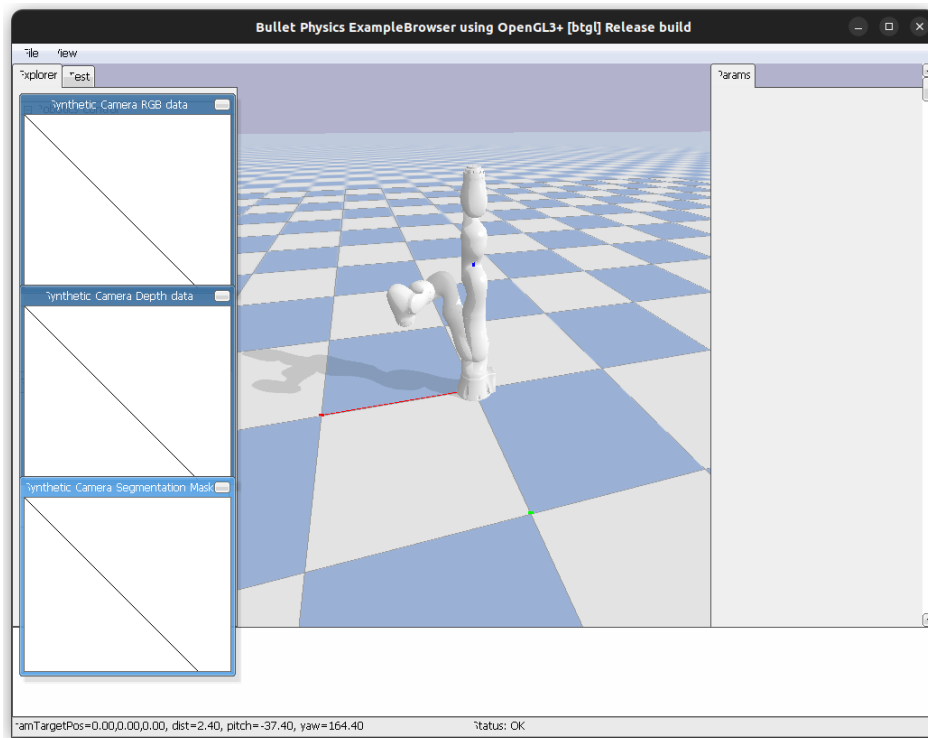


Figura 12: Posicion inicial Vs Posicion final del robot

## Conclusiones

El uso de **PyBullet** como motor de simulación física en proyectos de robótica y simulación ofrece múltiples ventajas, especialmente en entornos académicos y de desarrollo rápido. A continuación, algunas conclusiones :

- PyBullet permite una simulación realista de dinámicas físicas, incluyendo colisiones, fricción, y articulaciones robóticas, lo que facilita el desarrollo y prueba de algoritmos sin necesidad de hardware físico.
- Su integración con Python simplifica la programación y automatización de pruebas, convirtiéndolo en una herramienta accesible para estudiantes, investigadores y desarrolladores.
- La posibilidad de visualizar en tiempo real las simulaciones mediante interfaces gráficas, acelera la comprensión del comportamiento del sistema.
- PyBullet incluye modelos de robots, sensores y escenarios, reduciendo el tiempo de configuración inicial y permitiendo centrarse en el desarrollo del sistema o control.
- PyBullet tiene limitaciones en fidelidad gráfica y en comparación con simuladores industriales, por lo que su uso debe estar orientado principalmente a entornos de desarrollo, educación y validación conceptual.