Research on Simulation method

Using a simulation of our robot should help us solve real-world problems safely and efficiently.

The simulation should be as close to the real-life robot and its programming almost identical.

3D models:

· Model of the robot hand

Rotate in 3D( 2 motors).

Identify disk and their colours(2 optical sensors).

Pick up the disks from the belt and drop them in the appropriate place on the platform(maybe additional motor).

Play the nodes after sorting the disks(speaker).

Tutorial for building a movable robot model with URDF

(<https://wiki.ros.org/urdf/Tutorials/Building%20a%20Movable%20Robot%20Model%20with%20URDF>)

· Model of the conveyor belt system

Move in a constant speed

Have black and white disks on it

Have the robot connected to it

· Models of a black and white disk

· Model of the platform where the disk are placed/sorted

List of simulators:

| Simulator | Main Programming language | 3D rendering engine | Physics engine |
| --- | --- | --- | --- |
| Gazebo | C++ | ORGE | ODE/Bullet |
| SimSpark | C++, Ruby | Internal | ODE |
| Webots | C++ | WREN | ODE fork |
| OpenRAVE | C++, Python | Coin3D/OpenSceneGraph | ODE/Bullet |

Gazebo:<https://classic.gazebosim.org/tutorials> (general tutorials)

<https://classic.gazebosim.org/tutorials?cat=build_robot> (specific tutorials for robots)

Webots:<https://cyberbotics.com/doc/guide/tutorials> (general tutorials with 4-wheeled robot examble)

C++ libraries:

Github repository of libraries for optical sensors \*\*ARDUINO\*\*

<https://github.com/LRDPRDX/OpticalSensorsWithArduinov>

TensorFlow

To create a Gazebo model, we can use svg file <https://classic.gazebosim.org/tutorials?cat=model_editor_top&tut=extrude_svg>