

COZMO



([http](#))

BACKGROUND

- **Why remote access to a Cozmo Mobile Robot?**

You can remote access to the robot to control it locally from your PC, execute commands directly from your computer.

- **What are the examples of mobile robot for remote access?**

Remote access can be used to control the robot directly from the PC, monitor the surroundings and receive live feed via camera. It can have wide ranging applications surgery, diffusing bombs to let monitoring the surroundings through Cozmo's camera.

- **What are the pros and cons?**

The benefits of remote control are that the surroundings can be monitored via the webpage, and the robot can be controlled through the computer. The robot can independently explore local surroundings.

The cons are that due to the limited connectivity of Cozmo Wi-Fi, and required use of phone/ tablet as server engine to execute programs from workstation, the robot can only be controlled until a certain distance.

- **What are the objectives of this work?**

The objective of this work is to demonstrate possibilities with AI, computer vision, finite state automata upon integration with robotics. Additionally, the objective is to introduce students to python, calypso and code blocks as different tools to program the robot. The objective of remote control is to enable students to explore the possibilities with being able to control the robot locally being on their workstation.

OVERALL PROPOSED SYSTEM DESIGN

- **What does the system do?**

Cozmo is an AI enabled bot with in-built camera for computer vision. The system can be used to demonstrate AI, computer vision technologies, finite state automata and it can have multiple applications – it is a pick and place industrial robot with advanced capabilities to stack, roll cubes and make pyramids, speech generation and speech recognition, landmark based navigation.

- What is the overall proposed system architecture?

Cozmo Architecture

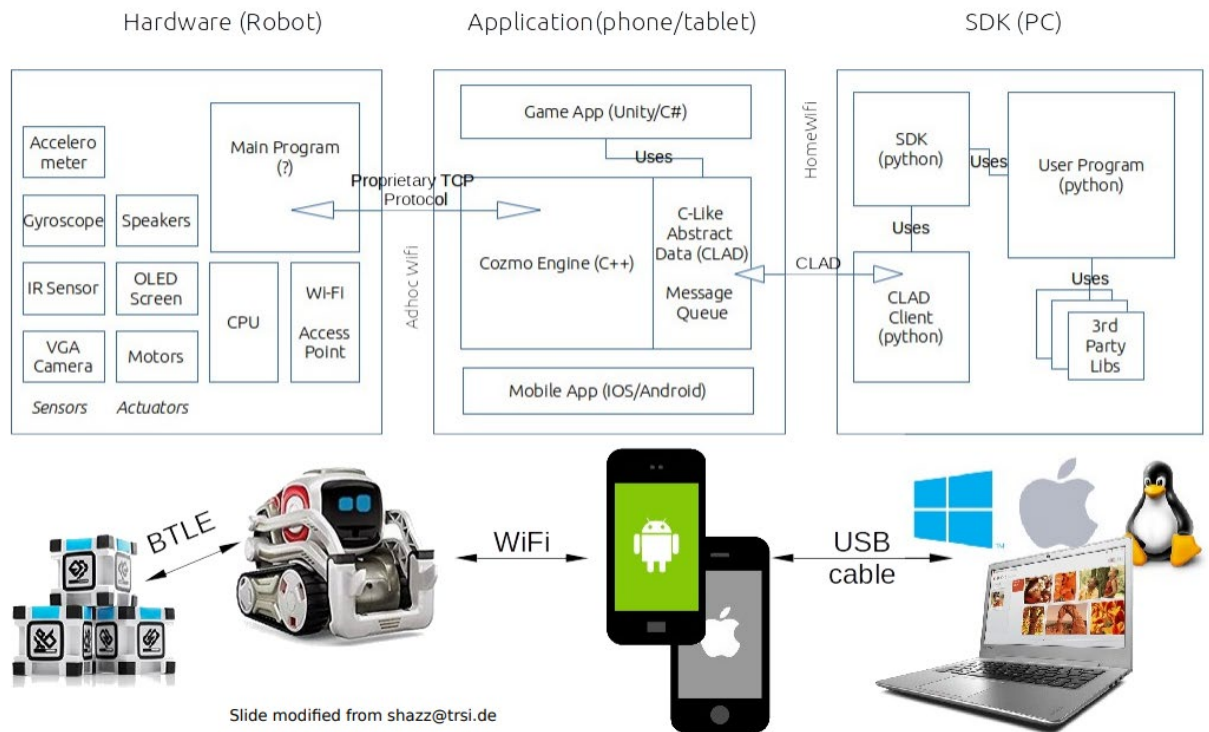


Figure 1: Cozmo Architecture

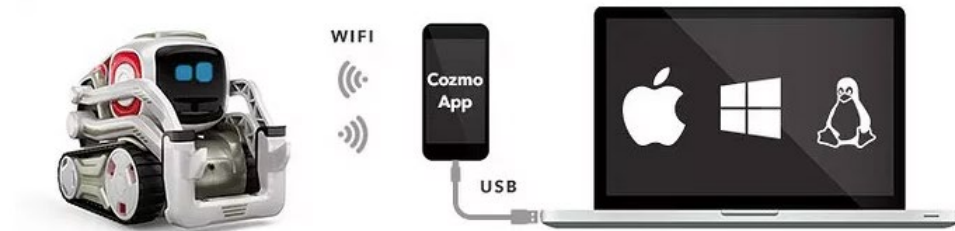
The robot consists of various sensors – accelerometer, gyroscope, IR sensor, VGA camera. The actuators are speakers, OLED screen, and motors. It has a CPU and Wi-Fi access point, and communicates to the application through proprietary TCP protocol. The application on the phone or tablet uses C-Like Abstract Data (CLAD), and the Cozmo engine is written in C++. Through the CLAD interface, the application connects through the homewifi to the PC, where user writes the programs using 3d party libraries. These programs are pushed to the application through CLAD client which uses SDK.

- What does the system consist of?

The system consists of:

1. Cozmo robot
2. Mobile phone/ tablet to run Cozmo app
3. Workstation to write and execute robot
4. Charging wire to connect the phone/tablet to the CPU

- **What are the basic functions of each system component?** (htt1)



Hardware setup for the Cozmo SDK.

Figure 2: Hardware setup for Cozmo SDK

Workstation to run the Cozmo SDK, write and execute the code

Phone to run the app in SDK mode and connect to the robot via Wi-Fi

Robot to execute actions according to the code.

WEB SERVER AND USER INTERFACE

- **Web server – what does it do? Which web server, does it do?**

“A web server is a system that delivers content or services to end users over the internet. A web server consists of a physical server, server operating system (OS) and software used to facilitate HTTP communication.

A web server is also known as an internet server.” (htt15)

“A web application or web app is any software that runs in a web browser. It is created in a browser-supported programming language (such as the combination of JavaScript, HTML and CSS) and relies on a web browser to render the application.” (htt14)

- **User interface – what does it look like? Why this layout? What can user do? Descriptions of each web page**

A simple UI designed for user to capture the world through Cozmo’s integrated camera. It is a simple layout with 2 different views – 1. To capture the surrounding’s through robot’s camera. 2. One camera to view the robot and view its movement.

This is a simple layout with basic functionalities integrated to execute commands on the robot. The user can make the robot move back, forward, right and left.

The user can toggle or play one of the 189 animations on the robot. The user can make the robot speak, make it move faster or slower, perform animations, or play freely.

COZMO SYSTEM CONTROLLER

- **Which one? Why? How many I/O points?**

Cozmo is controlled by the mobile app through 2.4GHz WiFi technology. The phone/ tablet on which mobile app is run is connected to the computer via USB cable with SDK mode enabled. This allows the user to write code/ programs on the workstation and execute them on the robot via the command prompt.

The only input point to execute the code on the robot is through enabling SDK mode on the app. The robot behaves in the way that the code commands it to do.

COSMO ROBOT AND SENSORS

- **What are the add on sensory components? What is the purpose of each sensory component?**

“Cozmo has a basic processor. Most computations that Cozmo makes actually happen in the cloud.” The cellphone app communicates with Cozmo and Anki servers.

“Infrared light and sensor on the bottom surface of his body is used for detecting objects.”

Physical sensors:

1. Camera : “Helps detect built-in objects, faces, and power cubes”
2. Cliff sensor: To prevent robot from falling down.
3. Accelerometer, Gyro : “Help the robot detect when it’s being hoisted from the ground”

SOFTWARE PROGRAMMING

- Web pages design – web server,

We use flask to host web server in python. It is minimalistic web application, and can be extended as necessary.

WHAT ARE THE APPLICATIONS OF COSMO ROBOT?

- **APPLICATION #1 : OBJECT DETECTION AND LANDMARK BASED NAVIGATION**

- **How does it work? using picture to show the layout**

It works through path navigation algorithm and Cozmo’s camera for computer vision and object mapping.

- **What does the code like?** The source code is from Anki’s official website

The code is pretty intuitive and easy to understand. It can be found in the downloaded examples from the ANKI website.

```

import cozmo

def cozmo_program(robot: cozmo.robot.Robot):
    lookaround = robot.start_behavior(cozmo.behavior.BehaviorTypes.LookAroundInPlace)

    cubes = robot.world.wait_until_observe_num_objects(num=2, object_type=cozmo.objects.LightCube, timeout=10)

    print("Found %s cubes" % len(cubes))

    lookaround.stop()

    if len(cubes) == 0:
        robot.play_anim_trigger(cozmo.anim.Triggers.MajorFail).wait_for_completed()
    elif len(cubes) == 1:
        robot.run_timed_behavior(cozmo.behavior.BehaviorTypes.RollBlock, active_time=60)
    else:
        robot.run_timed_behavior(cozmo.behavior.BehaviorTypes.StackBlocks, active_time=60)

cozmo.run_program(cozmo_program)

```

Source: From Anki Wesbite

cozmo_sdk_examples_1.4.10\cozmo_sdk_examples_1.4.10\tutorials\04_cubes_and_objects

- **APPLICATION #2: SPEECH GENERATION AND RECOGNITION**

- **How does it work?**

Natural Language Generation

In the quizmaster_cozmo the robot will ask single multiple choice questions with one correct answer

The user has to respond and then Cozmo's gives away the correct answer and moves to the next question

Refer to video 5 and 6 for demo [here](#)

- **What does the code like?**

cozmo_sdk_examples_1.4.10\cozmo_sdk_examples_1.4.10\apps\quizmaster_cozmo

- **APPLICATION #3: DR COZMO**

- **How does it work?**

State machines, event-based architectures, speech generation, landmark based object detection, facial recognition, computer vision, path planning algorithms, human-robot interaction, natural language dialog systems

Watch the example videos [here](#)

- **What does the code like?**
Basic programming in Calypso
- **APPLICATION #4 : INTRUDER ALARM**
 - **How does it work?**
State machines, event based architectures, Hough transform, face detection algorithms
 - **What does the code like?**
cozmo_sdk_examples_1.4.10\cozmo_sdk_examples_1.4.10\apps\desk_security_guard
- **APPLICATION #5 : MAKE A CUBE STACK**
 - **How does it work?** using the picture to show the layout
Object manipulation.
 - **What does the code like?**

```

import cozmo

def cozmo_program(robot: cozmo.robot.Robot):
    # Attempt to stack 2 cubes

    # Lookaround until Cozmo knows where at least 2 cubes are:
    lookaround = robot.start_behavior(cozmo.behavior.BehaviorTypes.LookAroundInPlace)
    cubes = robot.world.wait_until_observe_num_objects(num=2, object_type=cozmo.objects.LightCube, timeout=60)
    lookaround.stop()

    if len(cubes) < 2:
        print("Error: need 2 Cubes but only found", len(cubes), "Cube(s)")
    else:
        # Try and pickup the 1st cube
        current_action = robot.pickup_object(cubes[0], num_retries=3)
        current_action.wait_for_completed()
        if current_action.has_failed:
            code, reason = current_action.failure_reason
            result = current_action.result
            print("Pickup Cube failed: code=%s reason='%s' result=%s" % (code, reason, result))
            return

        # Now try to place that cube on the 2nd one
        current_action = robot.place_on_object(cubes[1], num_retries=3)
        current_action.wait_for_completed()
        if current_action.has_failed:
            code, reason = current_action.failure_reason
            result = current_action.result
            print("Place On Cube failed: code=%s reason='%s' result=%s" % (code, reason, result))
            return

        print("Cozmo successfully stacked 2 blocks!")

    cozmo.run_program(cozmo_program)

```

Source: From Anki website

cozmo_sdk_examples_1.4.10\cozmo_sdk_examples_1.4.10\tutorials\04_cubes_and_objects\cube_stack

- **APPLICATION #6 : ROLL A CUBE**

- **How does it work?** using the picture to show the layout
Landmark based object detection, object manipulation

- **What does the code like?**

\cozmo_sdk_examples_1.4.10\cozmo_sdk_examples_1.4.10\tutorials\04_cubes_and_objects\roll_a_cube

CONCLUSION AND FUTURE DIRECTIONS

Cozmo by Anki is one of a kind intelligent robot leveraging AI, machine learning, finite state automata to perform landmark based navigation, facial recognition and speech generation. It has a basic processor, but leverages cloud technologies through the connectivity to its mobile app. It is easy to code with programming capabilities in Python and free to download SDK. It can be used in surgeries, to help patients in hospitals, diffuse bombs and help in bakeries. It is one of a kind robot with unlimited possibilities unleashed by a combination of cloud, AI and machine learning combined with advanced path planning algorithms and kinematics

REFERENCES

- n.d. <https://www.cs.cmu.edu/afs/cs/academic/class/15494-s17/lectures/intro/intro.pdf>.
- n.d. <https://ankicozmorobot.com/cozmo-sdk/>.
- n.d. <https://anki.com/en-us/cozmo/cozmo-technology.html>.
- n.d. <https://www.microcontrollertips.com/teardown-anki-cozmo-vector/>.
- n.d. <https://www.microcontrollertips.com/teardown-anki-cozmo-vector/>.
- n.d. <https://www.microcontrollertips.com/teardown-anki-cozmo-vector/>.
- n.d. <https://www.microcontrollertips.com/teardown-anki-cozmo-vector/>.
- n.d. <https://www.microcontrollertips.com/teardown-anki-cozmo-vector/>.
- n.d. <https://www.instructables.com/id/When-COZMO-the-Robot-Meets-the-Raspberry-Pi/>.
- n.d. <http://heidloff.net/article/visual-recognition-for-cozmo-with-tensorflow>.
- n.d. https://www.reddit.com/r/Cozmo/comments/5yc70a/sdk_what_sensors_can_be_utilized/.
- n.d. <https://www.cs.cmu.edu/afs/cs/academic/class/15494-s18/lectures/intro/intro.pdf>.
- n.d. <https://www.cs.cmu.edu/afs/cs/academic/class/15494-s18/lectures/intro/intro.pdf>.
- n.d. <https://www.youtube.com/watch?v=eBvbLGOi8HM&t=614s>.
- n.d. <https://www.techopedia.com/definition/4928/web-server>.
- n.d. <https://calypso.software/>.