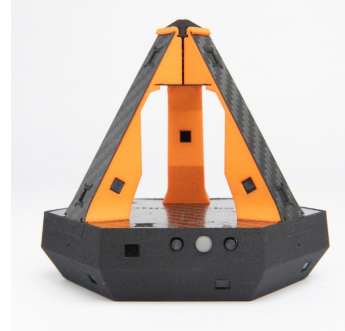


## Otus Tracker and RCbenchmark Tracking Lab

### Typical use

- Control of indoor UAVs and robots
- Dynamical characterization of robots
- Development of SLAM and vision systems
- Development of control systems and sensor fusion.



### Hardware features list

- Position, velocity, orientation, and rotational velocity measurement
- No calibration.
- Ultralight for mounting on UAVs
- Fast to install and portable
- Two input buttons
- Wireless communication
- Low latency
- Compact size: fits in a cube 10 cm wide.
- Small USB dongle for communication with the computer
- Wide field of view of the base stations to maximize tracking space in small rooms.
- Use up to 14 trackers in a single room.

### Technical specifications

**Table 1:** Design specifications of the RCbenchmark motor testing tool.

Specification	Value	Unit
Position precision	±0.5	mm
Position accuracy over whole tracking space	±5	mm
Orientation precision	±1	degree
Wireless communication rate	250	Hz
Pose update rate*	0.1 - 400	Hz
Latency	<4	ms
Weight	60	g
Supply voltage	4.3 - 5.5	V
Tracking volume	5x5x5	m <sup>3</sup>

\*The update rate is how often the Otus communicates with the computer. Using the optical information and the IMU readings, the computer can reconstruct the pose at a rate independent from the wireless communication rate.

**Software features list**

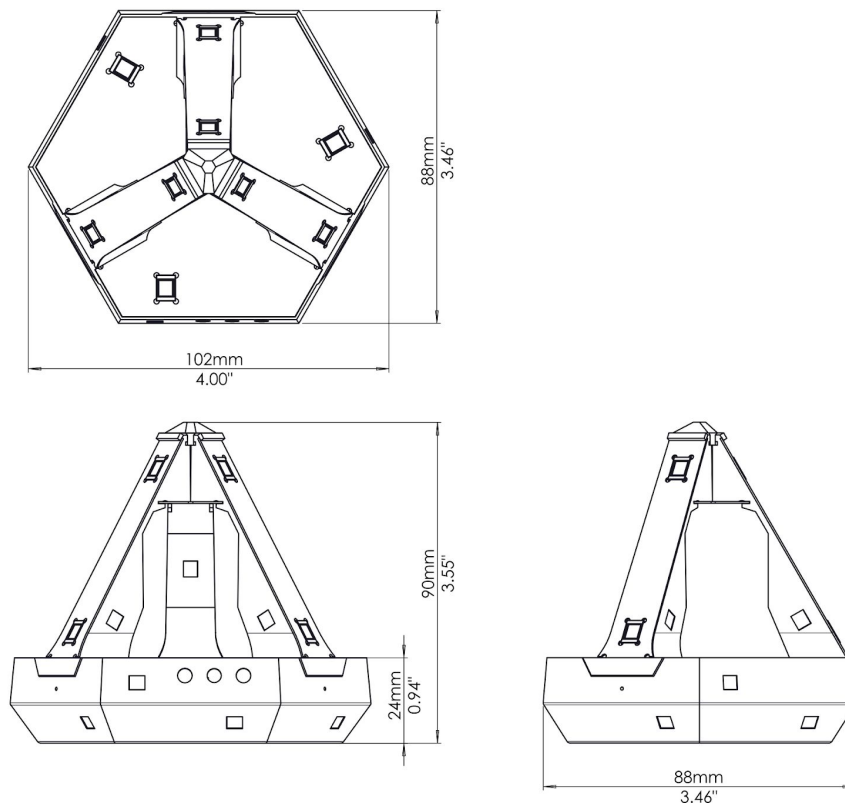
- Plugins for Matlab, C, C++, C#, ROS, and Python
- Simple integration with other languages supporting UDP
- Easy to use interface
- Export to CSV for analysis in almost any software
- Reset local axis with a single click
- Specify custom local axis for all controllers
- Uniquely identified controllers
- Prediction of future velocity and position to compensate for latency in control code

**Safety**

Be careful when testing flying vehicles and robots. They have the potential to cause serious harm or even death.

**Hardware**

The Otus tracker uses two base stations as tracking references. The base stations are passive emitters of infrared light. Please visit our website for installation [instructions](#). The dimensions of the Otus tracker are presented in figure 1. A CAD model of the Otus tracker is available [here](#) (FBX, OBJ and STL).



**Fig. 1: Otus tracker dimensions**

## Power

The Otus tracker is designed to be used on robots where weight is a constraint. Accordingly, the Otus tracker is powered by USB and does not include a battery. For onboard applications, a usb to female 0.1" three-pin header is provided. The header is the same size as a standard servo header.

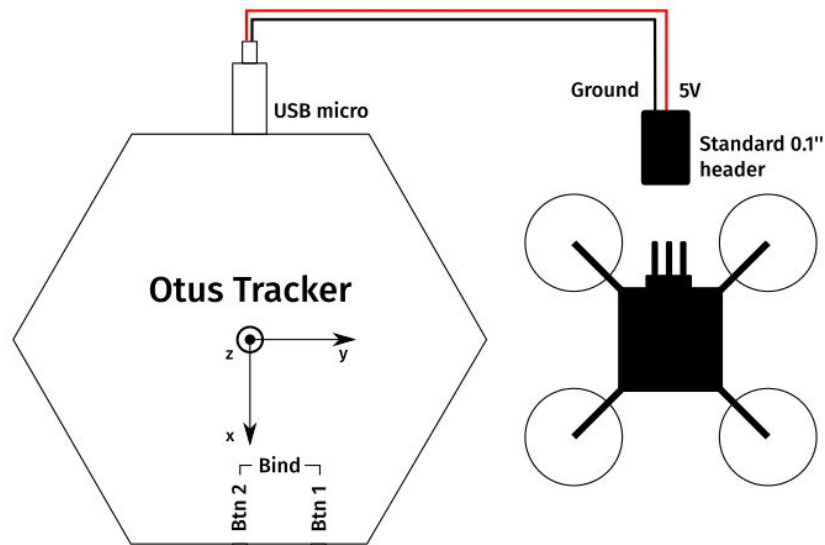


Fig. 2: Otus tracker dimensions

## Wireless connection

The Otus tracker system includes a small USB dongle. For better reception, we recommend increasing the distance between the computer and the dongle, as well as using the USB extension cable provided.

## Environmental condition for operation

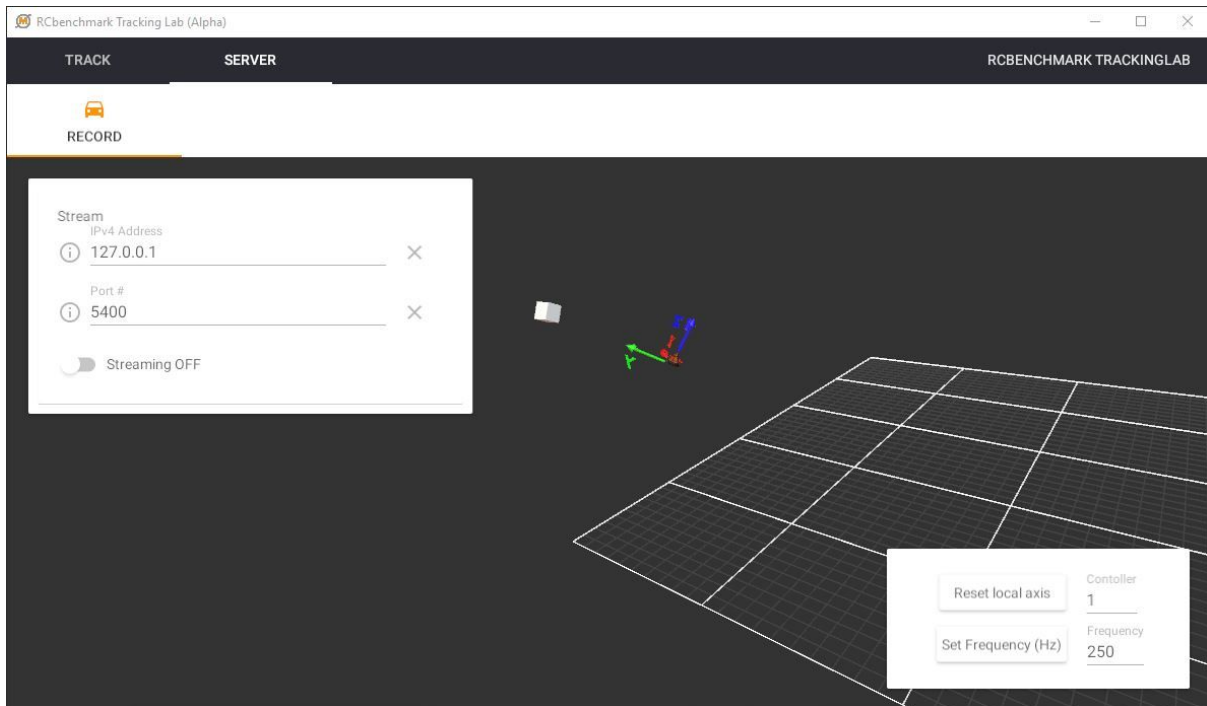
The Otus tracker is designed and tested for indoor use at typical room temperatures. Large mirrors will affect tracking accuracy and may cause loss of tracking. Outdoor use is not recommended as direct sunlight can overpower the sensors.

## Software

Minimum recommended specifications:

- Windows computer
- 4 GB of RAM
- i5 processor of second generation or better
- For smoother 3D rendering, we recommend a separate graphics card

The software is multithreaded to maximize performance and keep a fast update rate even on low power computers. We recommend using an i5 of second generation or better and 4 GB of RAM.



**Fig. 3:** RCbenchmark Tracking Lab interface

## Software feature

- Record to CSV file
- Offset center of coordinates
- Stream data via UDP
- Display pose information
- Set refresh rate
- Predict location of tracker in the future to compensate for controller latency.
- Reset local axis
- Display the controllers and base stations in a 3D environment

## Integration with other software

Multiple programming languages are supported and plugins are included for C, C++, Labview, Matlab, C#, ROS, and Python. Please visit the [download page](#) for more information. The software communicates with the UDP protocol. This means that the pose information can be sent to a program running on the same computer, or on another computer on a wired or wireless LAN network, as well as over the internet. The UDP packets are documented on our [website](#), so the software can communicate with any programming language that supports UDP.