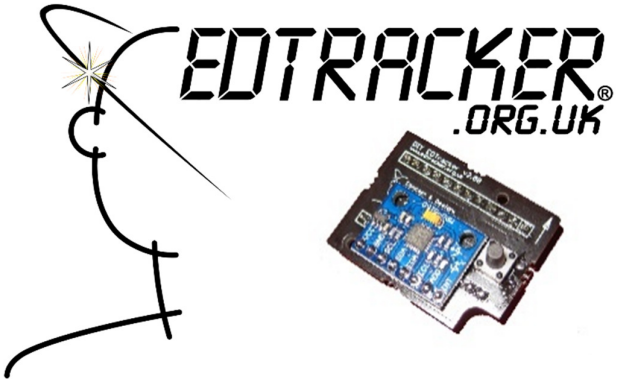


DIY EDTRACKER USER GUIDE

(MAGNETOMETER-ENABLED DEVICES ONLY)

Document Revision History

Version	Date	Summary
1.0	3rd Aug 2014	• Initial release for version 2 of the GUI
1.1	25th Sep 2014	• Updated for version 3 of GUI
1.2	6th Oct 2014	• Minor typos/corrections
1.3	10th Oct 2014	• Section for Experimental MPU9150 (Magnetometer) Support
1.4	22nd Oct 2014	• Formalised the MPU9150 instructions a little more, and fixed some minor typos/formatting.
2.0	22nd Jan 2015	• Split documents into two versions – magnetometer-enabled devices and those without. • Updated for version 4 of GUI • This document is MAGNETOMETER version
2.1	9th June 2015	• Updated to reflect minor changes in GUI
2.2	29th Feb 2016	• Updated for newer UI, firmwares, etc • Changes related to split of DIY EDTracker and EDTracker Pro, just for clarity

The definitive location for this document at time of writing is :-

http://www.edtracker.org.uk/index.php/downloads/category/3-documentation

1. Contents

2. Introduction

The EDTracker is a do-it-yourself electronic device that provides cheap, effective 3-axis head tracking for PC gaming. It uses a cost effective Invensense MEMS accelerometer and gyroscope coupled with the SparkFun Pro Micro development board to provide a small head-mounted device that can track the orientation of your head and reflect movement to a PC as a simple 3-axis joystick. This allows any software capable of accepting joystick input to track your head movements and represent that movement in-game.

The construction of the physical device is covered on the website (www.edtracker.org.uk). This manual assumes you have a constructed and working device.

No coding is required in order to make use of EDTracker. Our GUI software can flash your device with the necessary firmware provided you have built it using the standard components above. However, we understand the importance of open source and that some people may wish to tinker with the code and build it themselves – and that's absolutely fine too! The source code for the Arduino-compatible development board is open source and is also available via links on the website. The instructions for setting up a development environment, modifying and compiling your own version of the code are not covered by this manual, but are part of a separate guide on the website. The intention of this user guide is to cover the more common principles of getting your device up and working with the minimum effort.

EDTracker has been a great example of a community-driven project and we would like to take this opportunity to thank everyone who has helped out - in whatever capacity - for their assistance and support.

NOTE ~~PRO~~	These instructions are for home-built DIY devices only and are **NOT** related to the EDTracker Pro device, which is our commercial solution for those wishing to buy a fully complete device. If you are using a Pro device, please stop here and acquire the correct user guide for your device from the www.edtracker.co.uk website.

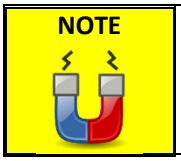
3. Disclaimer

(Hereafter, "EDTracker" refers to those individuals directly affiliated with EDTracker.org.uk and EDTracker Ltd)

The EDTracker software and printed circuit board (where used) is supplied on an as-is basis, without any warranties or support of any kind. DIY EDTracker does not come with any such warranty, whether oral or written, express or implied. By using the device, you agree to indemnify EDTracker Ltd from any losses, damages or expenses incurred as a result of your use of it. If any support or advice is offered by the EDTracker team, it is offered on an as-is, best-endeavours basis and does not constitute any formal agreement contrary to the above.

In short, plain English : this is not intended to represent a commercial-grade product and, since you built it yourself, don't blame us if it doesn't work, wipes your hard drive or gains self-awareness on August 29th and promptly sells your kids into slavery and destroys all your worldly belongings. Or anything else bad and nasty. You get the idea ☺. Right, with that out of the way…

4. Quick Start



These instructions are only applicable for devices equipped with a magnetometer. If your device uses an MPU-6050 board (non-magnetometer) please consult the *other* user guide available at :-
http://www.edtracker.org.uk/index.php/downloads/category/3-documentation

If you just want the summary of steps you will need to perform, without the detail, here's what you need to do. The detail is then provided in all of the following sections.

- Install the Arduino Drivers
- Download and launch the GUI
- Flash the firmware into the device
- Calibrate the gyro sensor offsets
- Calibrate the magnetometer
- Set up orientation, axis and sensitivity preferences
- Verify joystick operation
- Play

5. Hardware

The DIY EDTracker consists of an Arduino-compatible SparkFun Pro Micro development board (or clone), an Invensense breakout board (MPU-9150 or MPU-9250) and a tactile button to feature as a reset switch.

The Pro Micro board typically uses a Micro B USB connector to connect to your PC. A suitably long USB cable will be required. The USB connector on the SparkFun Pro Micro boards is a weak design and is prone to failure if undue stress is applied to the connector – you are advised to be careful when plugging/unplugging your device. Some stress relief on the cable is advised (see white cable tie in the picture below).

MPU-9x50 (Magnetometer)

In order to use a magnetometer-enabled EDTracker, please ensure you are using Version 4.0.4+ of the GUI as a minimum. Early GUI versions will not support the necessary features. The 9150/9250 makes use of a **single firmware** to perform both calibration and head tracking function.

Attaching to your head

Obviously we don't condone attaching the device *directly* to your head! Typically one attaches it to a pair of headphones or a hair band. Common approaches involve tape, adhesive Velcro tabs, rubber bands, zip ties or loom bands. Be aware that the device gets mildly warm during operation; this is normal.

Figure 1 - An EDTracker attached with an elastic band to a headset

NOTE 	Keep the device away from any strong magnetic fields. In particular, do **NOT** mount the device onto the earcups of headsets as the magnetic field from the speaker magnets invariable causes a problem.

NOTE 	While there are no serious voltages or currents passing around the device (it is 5 volt only), **do not** mount it directly to any metallic or conductive material without first insulating it. Alternatively, place it in a suitable enclosure.

Enclosures and Boxes

While not essential, placing your DIY EDTracker in an enclosure can make for a neater solution and can improve the performance of the device due to a more reliable, stable temperature. ABS plastic boxes are available from electronics suppliers, and the EDTracker PCB is designed to fit within a Hammond 1551GBK plastic enclosure with

minimal effort. It can be found from various electronics suppliers such as Maplin, Farnell and Mouser. Some people have reported good results from using a Tic Tac box! Alternatively if you have access to a 3D printer, some enclosure designs can be found on the website under the Hardware menu.

A word on USB cables

Not all USB cables are the same - please ensure it is a **data** cable and not a **charging-only** cable; the latter will power the device but will not provide suitable connectivity to the PC, and the PC will not recognise the DIY EDTracker. Some mobile phone cables are **charging-only**.

A word on USB ports

We've seen examples of the following. Please be aware of these issues and try alternatives should you think you have problems.

1. Some USB 3.0 USB ports can show issues if their drivers are not correctly installed/configured. If the device does not seem to work in yours, try a USB 2.0 port if you have one.
2. We have heard of mixed results when using the device through a USB hub or wireless extender. Try to plug the device directly into the PC whenever possible.
3. Windows 7's implementation of virtual COM ports over USB can be a bit odd, particularly when you've got a lot of devices. Clear out any old device drivers you don't need using a tool such as USBDeview (www.nirsoft.net/utils/usb_devices_view.html).
4. Some Windows 10 users report the device is not recognised after a reboot, unless it is unplugged and re-inserted. This appears to be specific to Windows 10.

A word on the button

We've had multiple cases where people have soldered the button in the wrong way, or have in some other way shorted out the button pins. When the main firmware is loaded into the device, holding the button down causes it to perform a continue calibration loop and it stops working as a virtual joystick. If you've built the device yourself, make sure you check the button operation!

If you wish to leave the button off your construction, this is OK. Leave the corresponding pins unconnected.

A word on temperature

Magnetometer-enabled (MPU-9x50) devices do not suffer from heat-related calibration changes like earlier non-magnetometer versions.

6. Software Installation

One of the first steps you will need to perform is the calibration of your device. This process ensures the device understands which way it is oriented and how the magnetic fields around it relate to its position.

Sketches, Images and Firmware

Your DIY EDTracker uses one primary component to perform the software function of head tracking – an Arduino-compatible development board, using a tiny Atmel microcontroller. You send the software into this microcontroller using a USB connection – the process of uploading the program into the device is called "flashing".

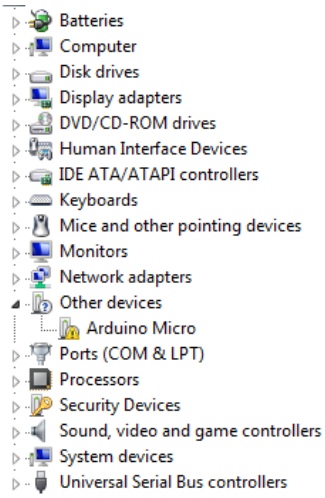
Depending on your viewpoint, the software that you flash into the EDTracker might be called any or all of the above. In true Arduino speak, a **sketch** is the name of the program you develop – typically some C code. You then compile that code into a binary **image** for flashing into the device. Other people may call this binary image **firmware**. For the purposes of this document, and to keep things simple, you may consider them all the same thing.

Drivers

Since the device is simply an Arduino-compatible microcontroller, it uses the standard Arduino drivers. These are available from the supplier of your Pro Micro or off the Arduino website as part of the Arduino software download. Alternatively, they can be downloaded from the EDTracker website, underneath the downloads section.

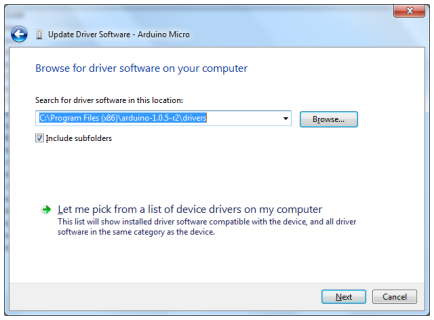
http://www.edtracker.org.uk/index.php/downloads/category/2-software

Unpack the ZIP file to a local folder. When asked by Windows for the device drivers, you should point it to this folder. If you wish to manually update the drivers for an existing device, perform the following steps.

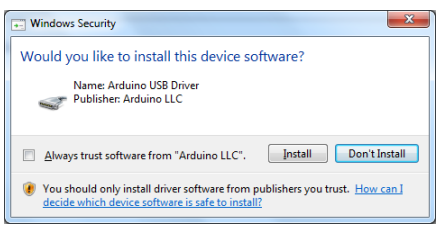


Click the Start button and right click on "Computer". Choose "Properties" to bring up the System window. Click "Device Manager" in the top left.

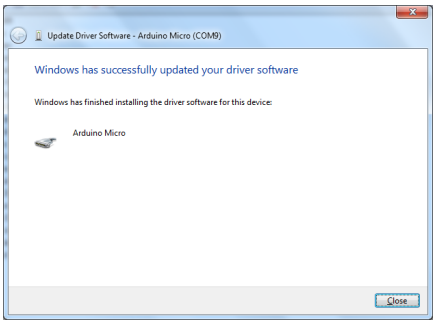
Find the "Arduino Micro" device with the warning sign on the icon, right click it and choose "Update Driver software". Click the "Browse my computer" option to locate the driver files on your hard disc (they're in the Arduino install directory)…



Navigate to the **drivers** subfolder of the Arduino install directory (e.g. C:\Program Files (x86)\arduino-*whatever*\drivers) and click **Next.**



Click "Install" – it's ok, we trust Arduino LLC ☺.



Hopefully you get the message on the left displayed. Your drivers are now installed. We're good to go!

DIY EDTracker GUI

The DIY EDTracker Graphical User Interface application is used to flash and calibrate your device. It can be downloaded from the EDTracker website (www.edtracker.org.uk) under the Downloads → Software section.

Download the GUI and unzip it to a local folder on your PC. Ensure you have installed the Arduino drivers before using.

The GUI automatically connects to the internet and retrieves available firmwares for your device. Ensure any firewalls allow the application to access the internet in order to flash the device.

http://www.edtracker.org.uk/index.php/downloads/category/2-software

!	**The EDTracker UI requires Windows XP or later and .Net 4.0 or later http://www.microsoft.com/en-gb/download/details.aspx?id=17851**

Flashing Firmware

With your device connected and the DIY EDTracker GUI running, you should see your device listed in the "Program EDTracker" section. Typically it will be listed as "COMn – Arduino Leonardo" or "COMn – SparkFun Pro Micro". If it cannot be found, click "Scan Ports" to rebuild the list.

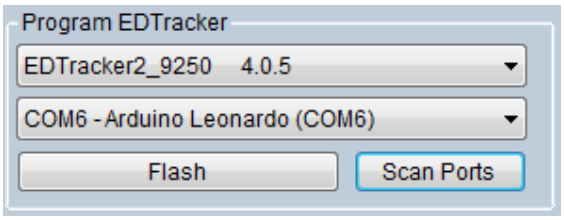


Figure 2 - Programming the EDTracker

Available firmwares are listed in the first drop-down box – a firmware name (e.g. "EDTracker2_9250") and a version number (e.g. "4.0.5").

Pick the latest firmware name that relates to your physical hardware – either a MPU-9150 or MPU-9250 build.

Click "Flash" to flash the firmware into your device. Do not disconnect the device during the flash procedure.

7. Initial Calibration

The MPU9x50 sensor comes equipped with a magnetometer. This enables the elimination of yaw drift over extended periods of time but it is essential to calibrate the magnetometer in order for it to function accurately.

NOTE !	**The 9x50 firmware always performs an automatic 20 second calibration on power-up.** To reduce the need for a separate calibration sketch the 9x50 firmware calculates gyro bias values on start-up. The EDTracker **must** be kept level (button facing upwards) and stationary for that period. A Full Bias Calibration can also be triggered from the UI.

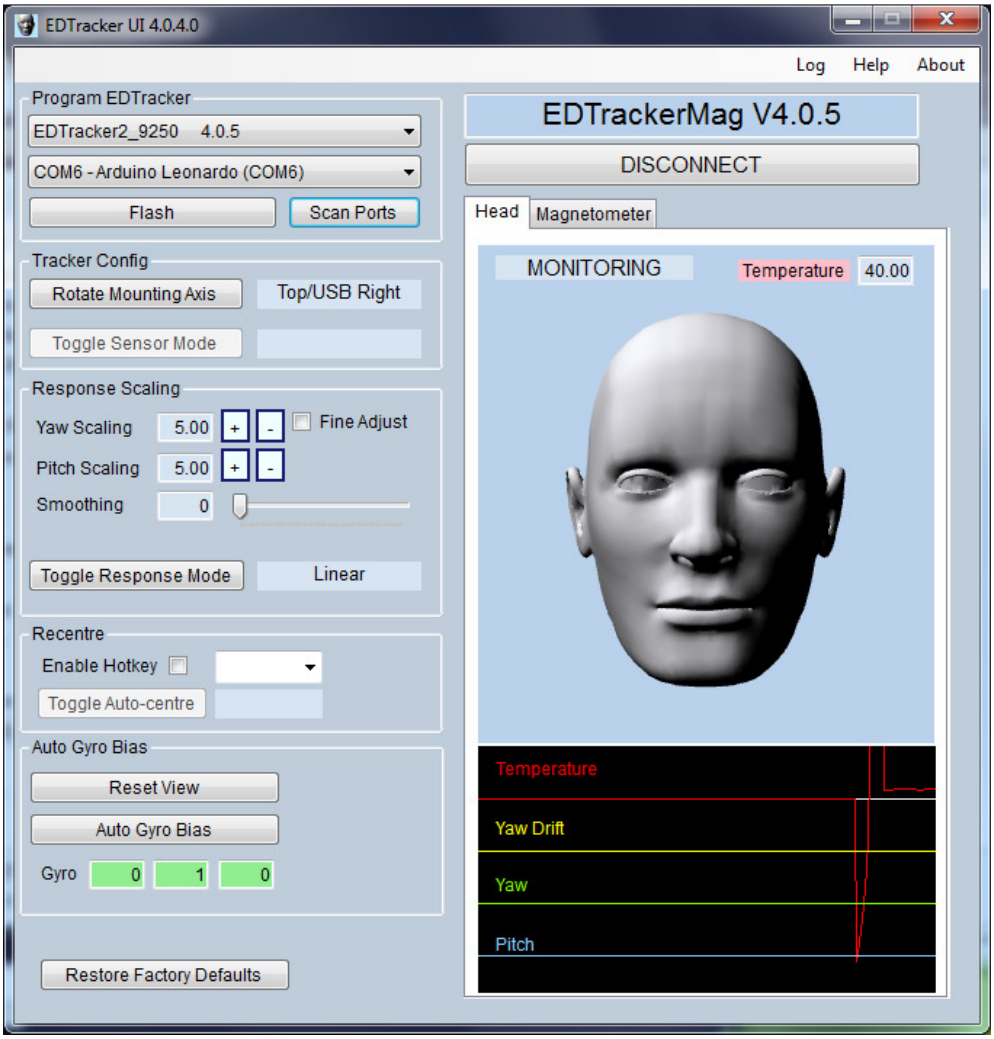
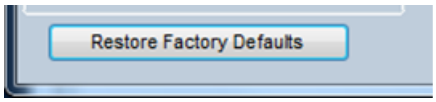


Figure 3 - The UI with a 9250 EDTracker connected

Initial Commissioning

If you have built your own 9x50 EDTracker then you will need to first select the EDTracker2_9x50 sketch from the drop down list and click Flash. Keep the EDTracker stationary until the 20 second Auto Bias Calibration completes.

Then click the Restore Factory Defaults button at the lower left of the UI.



The head will wobble around at this point, which is fine.

Magnetometer Calibration

Select the Magnetometer tab just above the head. The initial uncalibrated state looks as follows:

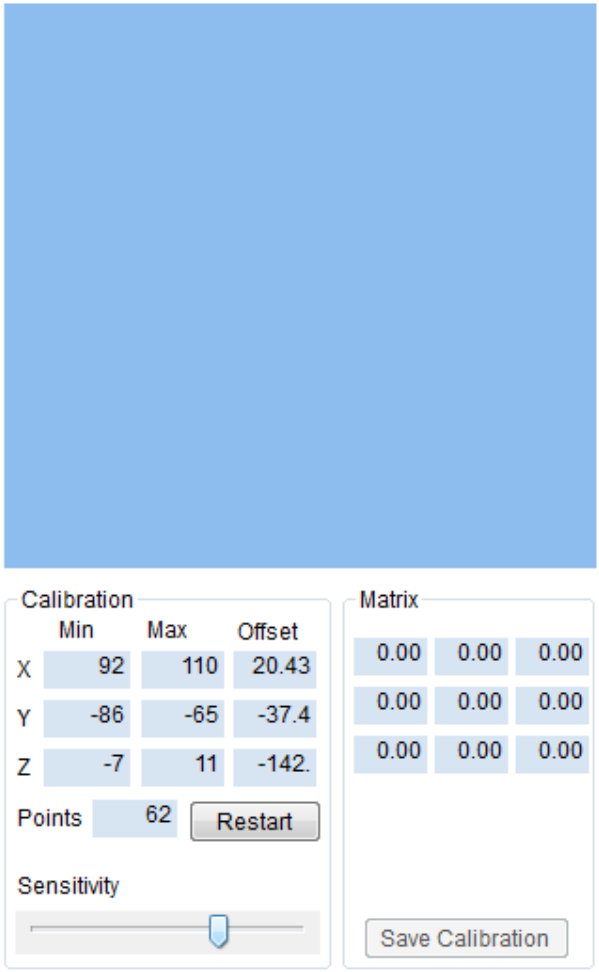
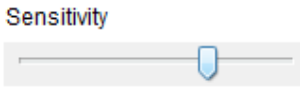


Figure 4 - Magnetometer Tab

In the lower left are the currently known minimum and maximum values read from the magnetometer. The values running vertically on the right hand side represent the transformation matrix which will correct magnetic distortion.

Adjust the Sensitivity slider to the right until the Points count starts to increment (typically around 75% across)



Rotate the EDTracker in multiple directions, around all 3 axes, until at least 500 Points have been sampled. Try to cover all possible orientations of the device.

Click “Save Calibration”.

NB: This video demonstrates how to get a good sample - http://youtu.be/-CRCJloZgYQ

Two sets of blobs will be displayed, forming rough spheres. The red blobs show uncalibrated samples. The green blobs show the calibrated samples.

The green blobs should form a rough sphere, centred on the display, while the red sphere will be off-centre and may well be distorted in shape.

If the green samples are not uniform or contain spikes then click Restart and perform the calibration again.

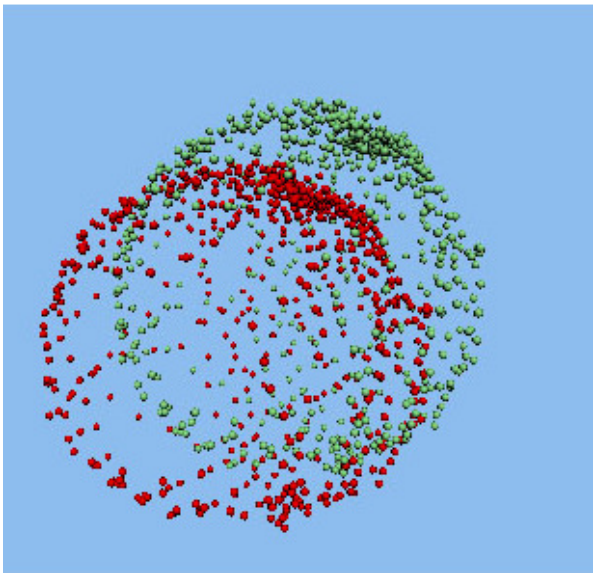
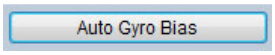
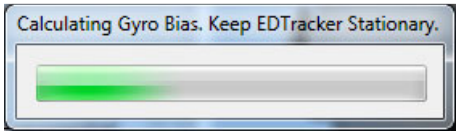


Figure 5- Magnetometer Samples

Return to the main Head tab and with the EDTracker **stationary and level** click ‘Auto Gyro Bias’.



A pop-up will appear indicating that calibration is in progress. Do NOT move the device during this period. If you do, let it complete and then restart the auto calibration process.



After 20 seconds the head should stabilise.

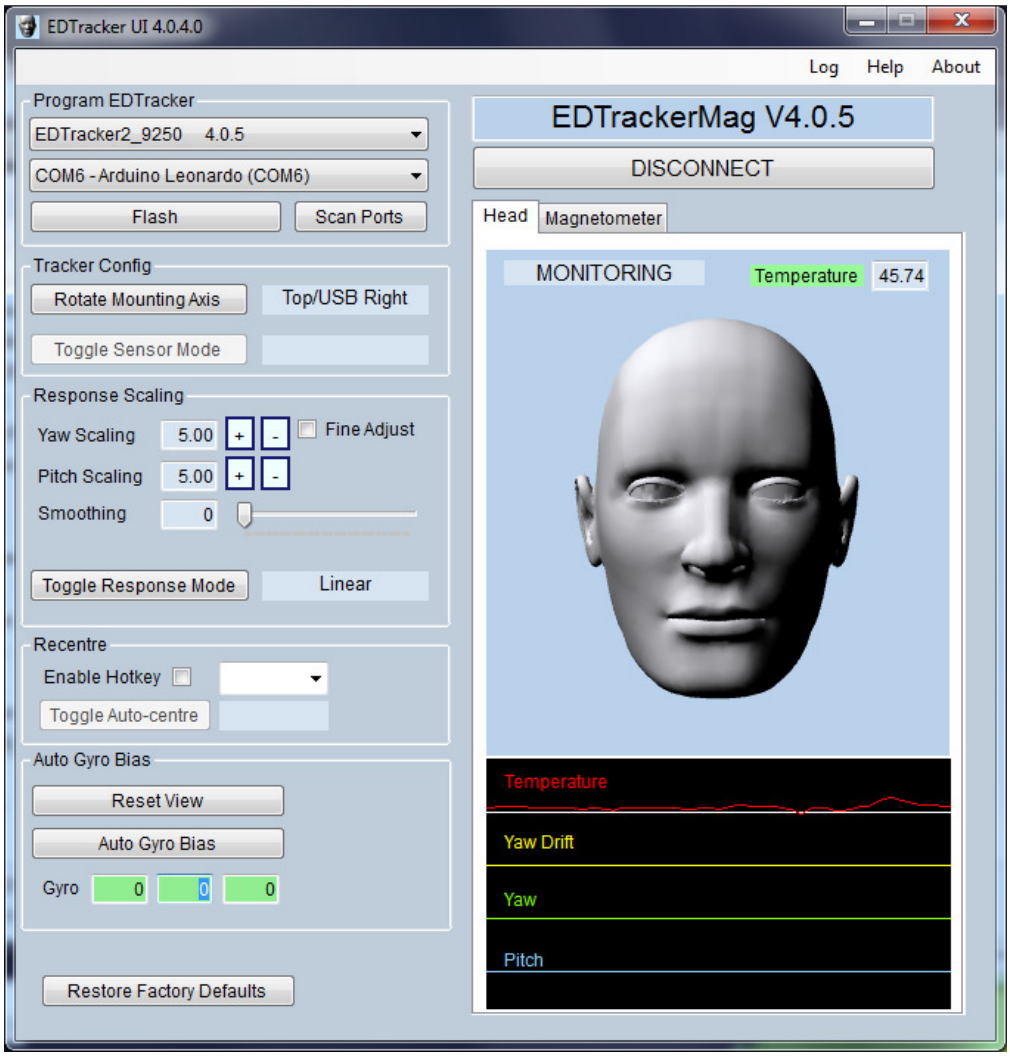


Figure 6 - A fully calibrated EDTracker

Note that with the device flat and still, the three “Gyro” values should all be approaching zero and in the green (it is normal for them to fluctuate slightly). If you move the device you will see these move.

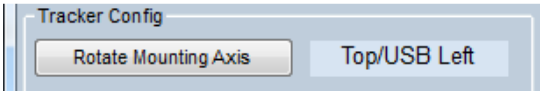
8. User Preferences

After magnetic and gyro calibration has been performed, you can tailor the configuration of the EDTracker to your individual preferences using the following options:

Device Orientation

NOTE !	The 9150/9250 EDTracker must be top-mounted only. Side mounting is not currently supported.

You need to set the orientation in the GUI to coincide with the physical orientation of the device. By clicking the ROTATE MOUNTING AXIS button, you can cycle between various descriptions of the device position. Check the movement of the device with the movement of the on-screen head, and cycle the option until it matches.

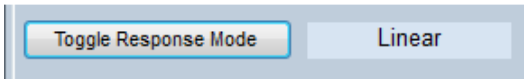


If you are using the EDTracker PCB design, the description below the button should hold true. For example, “Top/USB Right” means the device is on the top of your head, with the USB cable coming out to the right. If you have custom-built the device with your own design, you may need to experiment with available orientations.

You may find that after changing orientation you need to re-perform the gyro calibration.

Response Mode

The device can output movement in one of two modes – linear or exponential. You can cycle between alternate modes by clicking the TOGGLE RESPONSE MODE button.



Exponential mode gives reduced responsiveness around the straight-ahead position, but the head movement then ramps up as you move off-centre. Linear is a more conventional mode where the responsiveness remains the same across the whole range of movement.

Note that a dead-zone can usually be set in-game if you prefer to have no view motion when looking dead ahead.

Axis Sensitivity (Scaling)

You can adjust the sensitivity of the X and Y axis (yaw and pitch) using the + and – buttons next to the “Yaw Scale” and “Pitch Scale” values.

Higher values give more in-game movement for physical head movement. By enabling the “Fine Adjust” checkbox you can make smaller (0.25) adjustments.

Smoothing

Smoothing adjusts the sensitivity of the EDTracker to small movements. This can be used to reduce the jitter effect of involuntary head motion. A value of 0 gives no smoothing and sharpest response. It is required less within exponential mode, but can improve linear mode at the expense of responsiveness.

Re-centre Hot Key

You can immediately re-centre the view in-game by pressing the assigned Function Key specified here. e.g. F9

9. Using in-game

Joystick

The DIY EDTracker pretends to be a 3-axis joystick; that is all. So any games that allow you to map head look movement to an analogue joystick should work – just set the joystick axes accordingly.

- X axis is left/right movement of your head (yaw)
- Y axis is up/down movement of your head (pitch)
- Z axis is rotational movement of your head (roll) i.e. Tilting your head to the left or right side

Do not perform any calibration of the device in your windows USB Game Controller settings. This may interfere with its use. If you do this by mistake, you can reset the calibration to defaults using the Advanced properties within Windows USB Game Controllers.

Position Reset during gameplay

During gameplay, if you find the device has wandered off-centre you can "zero" all the axes back to straight-ahead by configuring and using the "Re-centre" hot key in the GUI. With the device in position on your head, look straight ahead at your screen (ie. Hold your head in the "straight ahead" position) and press the defined function key. The device will re-calibrate all axes to zero at this position.

Quick Recalibration

Pressing the button on the device (or re-plugging the device into a USB port) will initiate a short calibration and reset the device to dead-ahead.

Game Support

EDTracker should work with any game that supports an analogue joystick for headlook input.

In addition, for those games that only support native head tracking protocols such as FreeTrack, TrackIR and FSX, we can highly recommend the software "Opentrack" by Stanislaw Halik. Instructions for setting up EDTracker with Opentrack are at the following link :-

http://www.edtracker.org.uk/index.php/using/opentrack

10. Common Issues

View drifts off to the left or right with no head movement

If you are seeing very rapid drift to the left or right (but vertical axis performs as expected) then it is highly unlikely the sensor is faulty – in 99% of cases this is incorrect magnetometer calibration. Re-tread your steps in the previous section slowly and verify you've done it right.

In rare cases some metallic headsets are made of a ferrous material that may be interfering with the magnetometer of the device; try the device *off* the headset as a temporary measure and see if the behaviour continues. If not, this may suggest your headset is interfering with the device.

Looking up or down introduces sideways movement

The orientation value is incorrect for how your device is mounted. Possibly the device is mounted upside-down, Check your orientation setting. If you've recently updated the GUI or firmware on your device, reset the device to factory defaults and then repeat the calibration steps in this document.