EDTRACKER

DIY Head Tracker for PC gaming

EDTRACKER USER GUIDE

(NON-MAGNETOMETER 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 NON-MAGNETOMETER version.

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

http://www.edtracker.org.uk/index.php/downloads

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 Arduino 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.

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 is not covered by this manual, but is part of a separate guide available upon 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.

3. Disclaimer

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

The EDTracker software and printed circuit board (or the complete pre-assembled device, where supplied by the EDTracker development team) is supplied on an as-is basis, without any warranties or support of any kind. EDTracker does not come with any such warranty, whether oral or written, express or implied. By using the device, you agree to indemnify EDTracker 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, whether we built it for you or 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

NOTE	These instructions are **NOT** applicable for devices equipped with a magnetometer. If your device uses an MPU-9150 board (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.

"I built the device myself from scratch"	*"I've got an existing device built up for me"*
• Launch the GUI • Flash the calibration firmware into it • Let the device warm up • Calibrate the sensor offsets • Flash the main firmware into it • Calculate drift compensation by leaving the device over a period of 5+ minutes • Set up orientation, axis and sensitivity preferences • Verify joystick operation • Play …in the future, you can repeat the configuration steps as needed.	• Launch the GUI and configure the device for correct orientation, axis and sensitivity settings (as per your preference) • Play • If you find it is drifting to the left or right quickly, perform the recalibration steps outlined on the left

5. Hardware

The EDTracker consists of an Arduino-compatible SparkFun Pro Micro development board (or clone), an Invensense breakout board 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.

MPU-6050 MEMS Accelerometer & Gyroscope

The MPU-6050 device does not include a magnetometer, therefore it is important that you flash the correct firmware into the device – attempting to use the MPU-9150 version of the firmware in a non-magnetometer device will not cause any damage, but will not function correctly.

Attaching to your head

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

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

You can attach the device on the side also, but the device will need appropriate configuration (this is covered in detail later).

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 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 latest 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 EDTracker. Some mobile phone cables are typically **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).

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!

A word on temperature

The accuracy of the device can be affected by large changes in temperature (roughly ±5°C). Allow your device to warm up before calibrating and also before gameplay; 5 to 10 minutes usually suffices. Your device may need periodic recalibration, for example between winter and summer months. The temperature of the device is displayed in the GUI, along with a red graph line showing if it is increasing or decreasing in temperature. Typical temperatures for a device in an enclosure are ~32deg C.

6. Software Installation

If you have received a pre-built device, then it will work as a virtual joystick "as-is" without any need for drivers. However if you want to adjust any of the settings (sensitivity, mounting position, etc) you will need to install the drivers and software to change them.

If you have built it yourself or you need to calibrate or flash a device, then you will need to install the relevant drivers and GUI software also.

Sketches, Images and Firmware

Your EDTracker uses one primary component to perform the software function of head tracking – an Arduino 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".

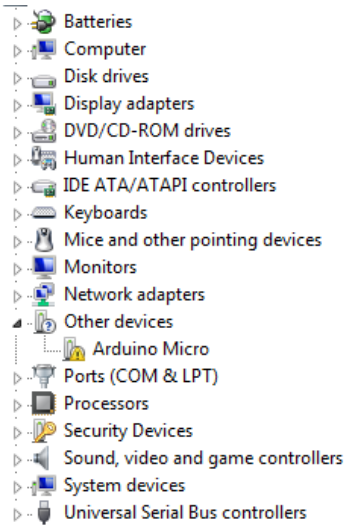
Depending on your viewpoint, the software that you flash into the Arduino might be called any or all of the above. In true Arduino speak, a **sketch** is the name of the program you develop – typically a load of C code. You then compile that code into a binary **image** for flashing into the device. Other people may call this binary image **firmware**. For the purposes of this document, and to keep things simple, you can 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 off the Arduino website as part of the Arduino software download. Alternatively, they can be downloaded from the EDTracker website, underneath the downloads section.

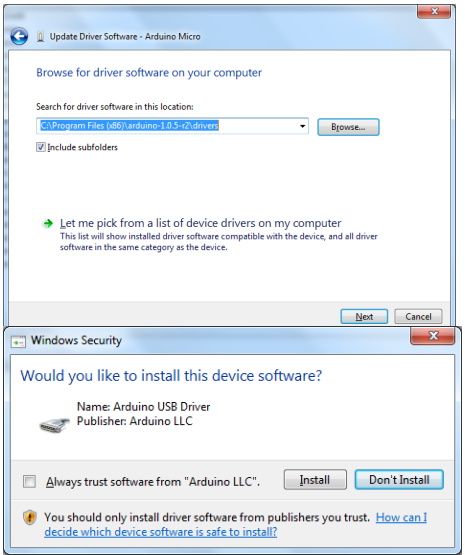
http://www.edtracker.org.uk/index.php/downloads/category/2-software

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.



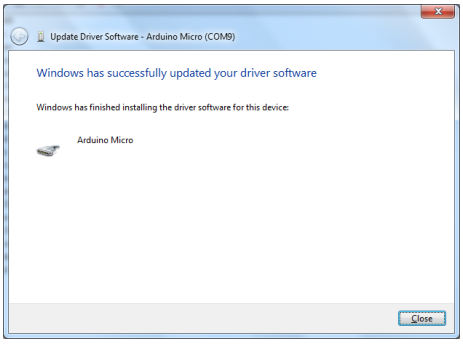
Click the Start button and right click on "Computer". Choose "Properties" to bring up the System window. Click "Device Manager" in the top left.

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!

EDTracker GUI

The EDTracker Graphical User Interface application is used to flash and calibrate your device. It can be downloaded from the EDTracker website 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 devices. 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

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

7. Initial Calibration

NOTE: The following instructions are for the non-magnetometer version of the software.

If you purchased a pre-built device from EDTracker.org.uk, this step will already have been performed. If you find that drift is no longer manageable using 'Drift Compensation' (see later section) then it can help to perform this step yourself.

Plug the device into a free USB port and mount it somewhere flat and still.

Launch the EDTracker GUI application by double-clicking ***EDTrackerUI4***.exe You should be presented with a screen similar to the following (if your device is not blank, it may look slightly different – the below is for a completely blank device that has just been assembled, but does not have any firmware loaded into it).

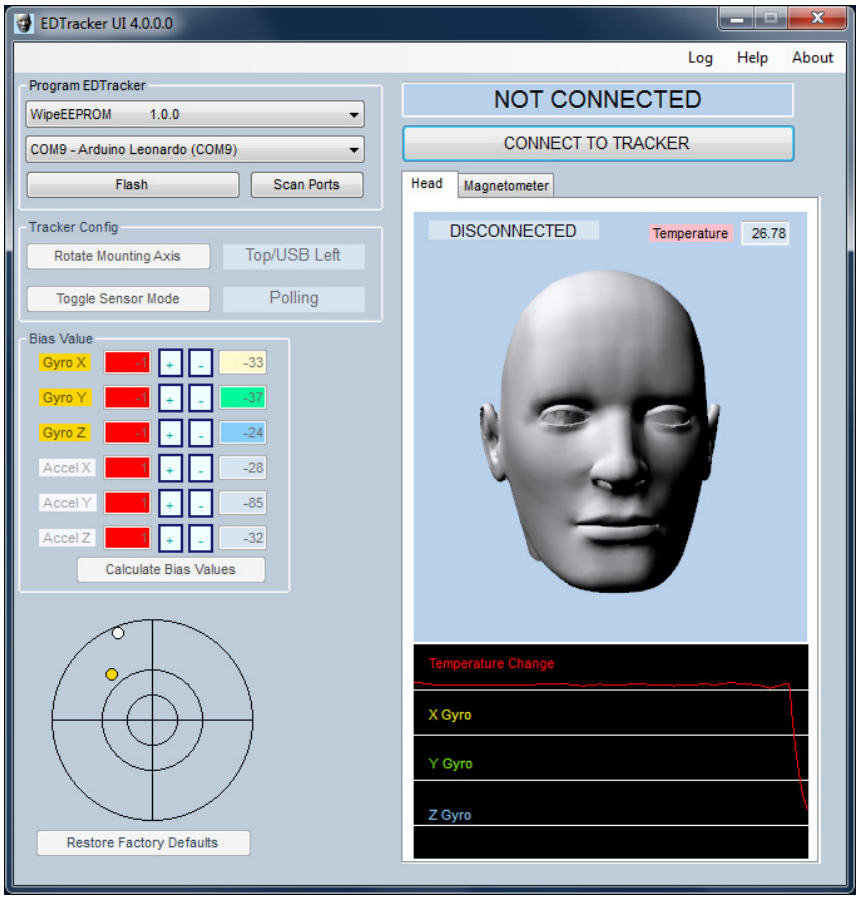


Figure 2 - Application screen with no recognised device

In the 'Program EDTracker' section (top left), select the latest "**EDTracker2Calib**" image – the version numbers are shown in brackets; the highest number is the latest. In the above example we are choosing version 2.5.3. Then select the PORT where the device appears (it will show as an Arduino Leonardo or similar).

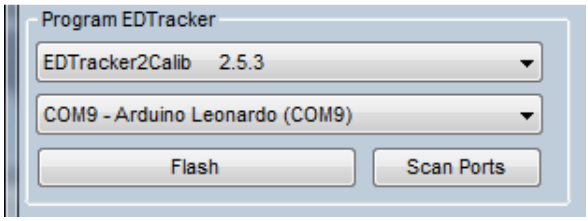


Figure 3 - Firmware selection and COM Port list

Click the Flash button. A window will pop up showing progress as the selected code is written to the EDTracker.

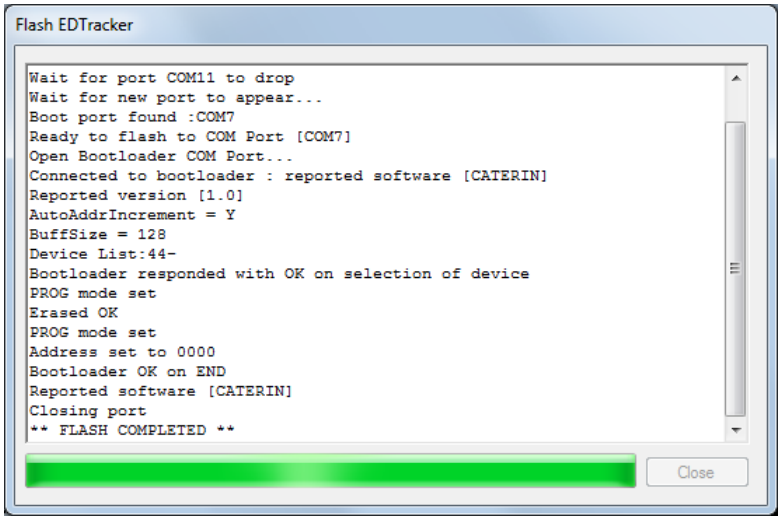


Figure 4 - Flashing the device

Occasionally an error will be reported by this window. Close the window and If the selected sketch does not appear to be running the flash process can be retried.

Once flashed, the device should be re-detected and the screen should change to something similar to the following…

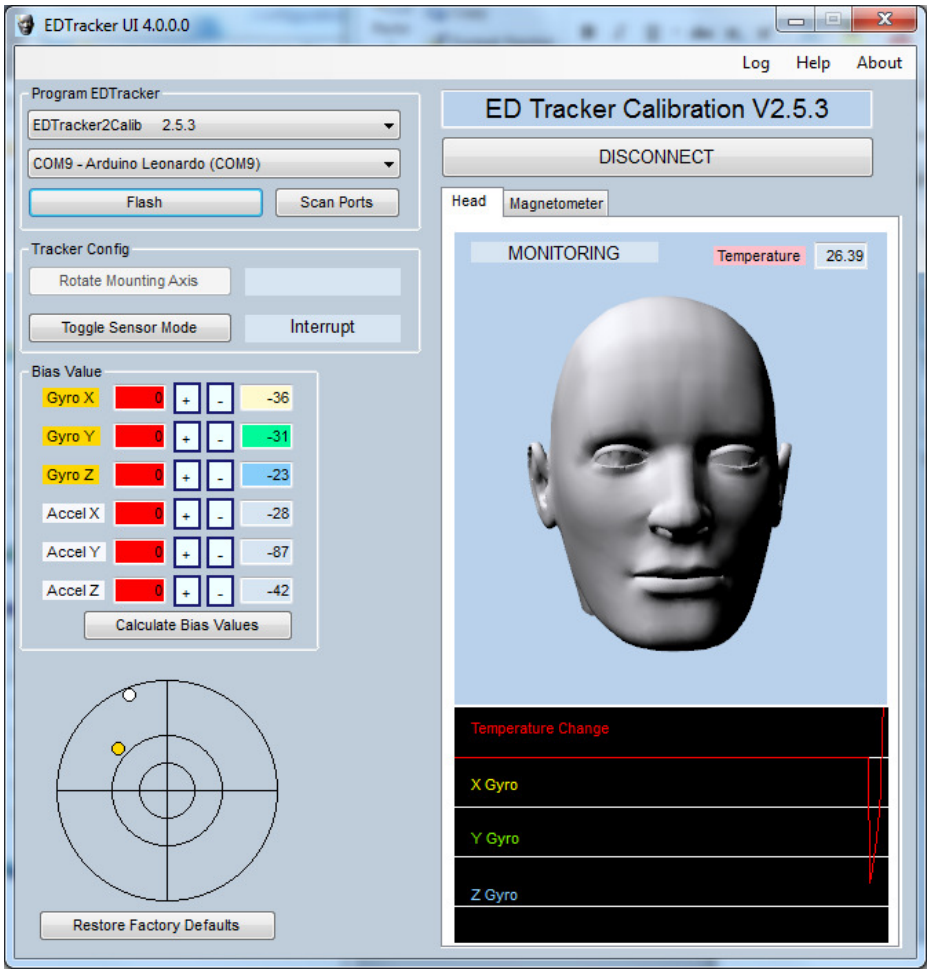


Figure 5 - Application screen with Calibration image loaded into device

Wait until the red line (temperature) has stabilised; this shows that the device has warmed up. Temperature affects the calibration so try to get the device up to normal operating conditions first. The Temperature label to the upper right of the 3D Head will become green when temperature has stabilised.

If this is a brand new device then select ‘Wipe Tracker Settings’.

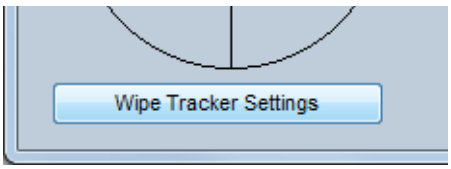


Figure 6- Wipe Tracker Settings

NOTE !	**During the following steps it is VITAL that the device remains completely flat and still.** Clamping the device with a suitable tool, taping it to or holding it under some books are all good suggestions. Failure to hold the device flat and still will result in poor calibration, and poor in-game performance.

When you're happy the temperature has stabilised, click the "Calculate Bias Values" button. After a short period the orange and white dots on the crosshairs should move towards the centre. You want to get them within the inner circle of the crosshairs. This may take several clicks of the 'Calculate' button, particularly if you knock or move the device. **Re-click the button as often as you need**.

When the two dots are within the inner circle, that's good enough. You can make fine adjustments to the values using the + and – buttons until all Gyro and Accel values are zero or close to zero (showing as green).

All changes are applied immediately to the EDTracker which saves them to memory.

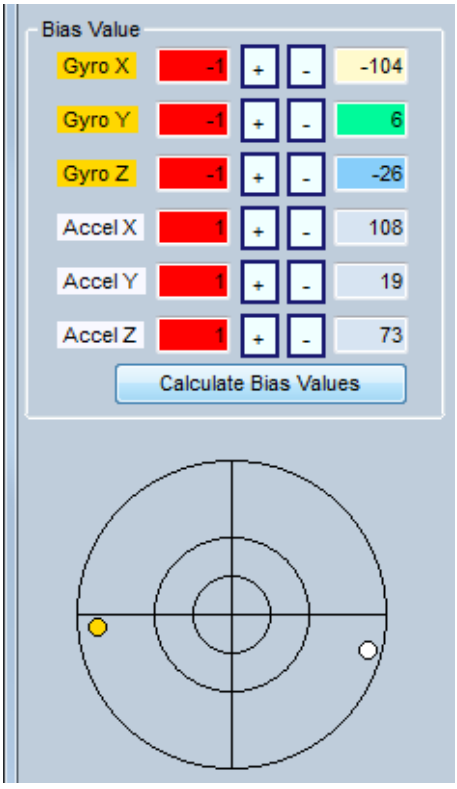


Figure 7 - Uncalibrated Device

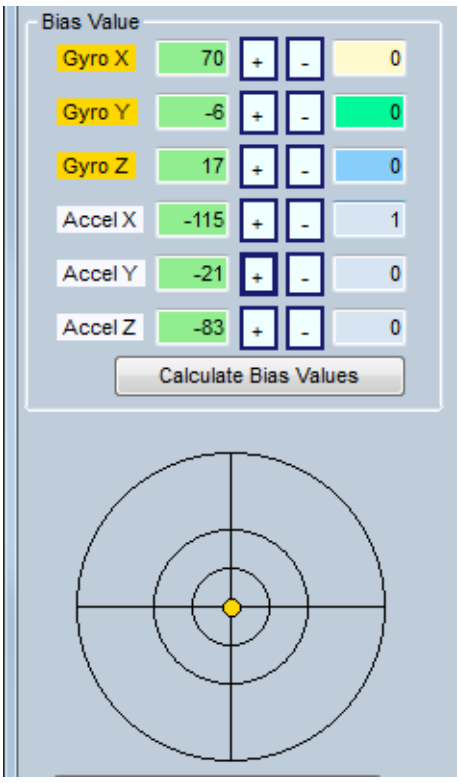


Figure 8 - After calibration

8. Head Tracking Operation

In order to function as a head tracker, the device needs the "main" EDTracker2 image loading. If you have bought a device pre-assembled, it should have this already performed. If you have built the device yourself, you must perform this step **after** you have performed the previous calibration activity. When running the main EDTracker firmware you can also perform the final 'drift compensation' procedure.

Flash main firmware

Start the GUI (if it is not already running) and select the latest version of the main EDTracker2 firmware from the drop down list. Generally pick the version at the top of the list (with the largest version number).

Make sure you choose the EDTracker2 image, **not** the one ending with Calib (these are calibration images).

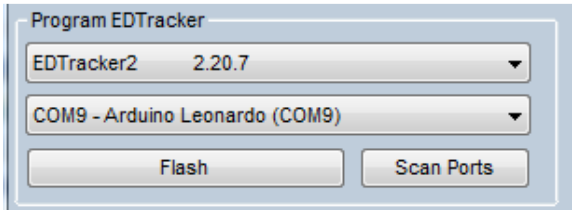


Figure 9 - Flash the main EDTracker2 image

Once you have flashed in the main firmware and with the device still plugged in, the menu options will now differ slightly, and the UI will show monitoring of an "EDTrackerII V*x.x*" device.

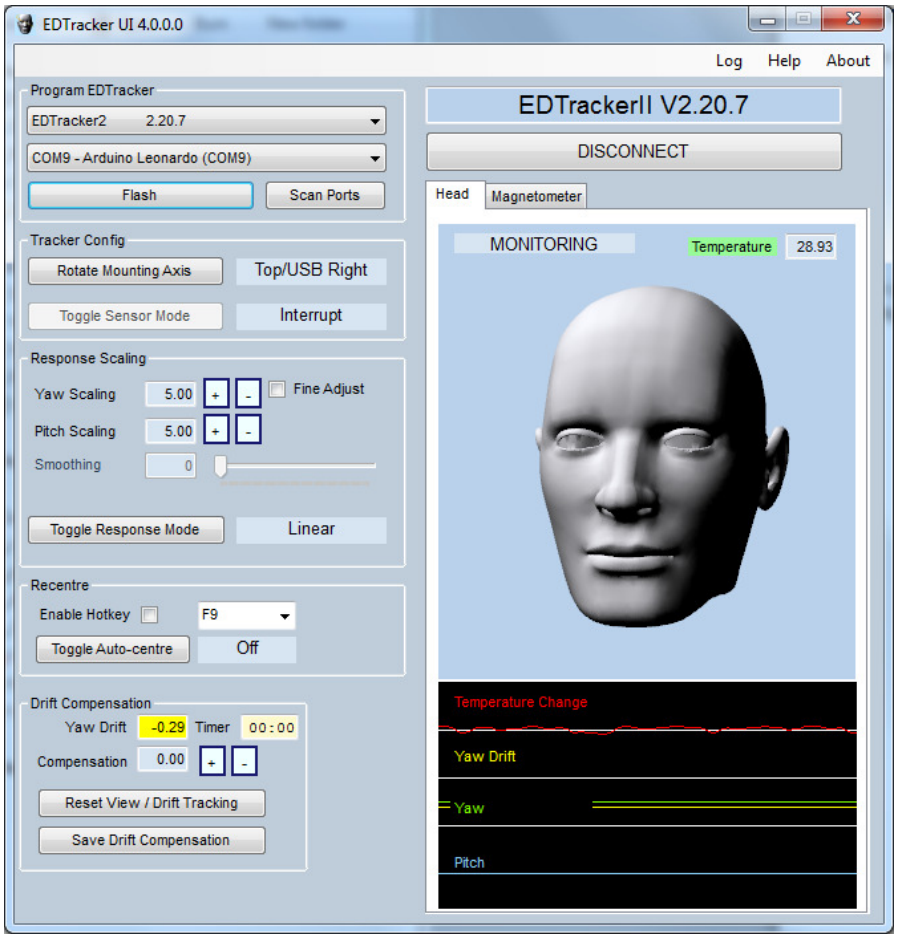


Figure 10 - GUI when connected to the main EDTracker2 firmware

Calculate drift compensation

What is drift?

In order to make the device more accurate over extended periods of time, we are going to leave it to calculate how much it drifts over time. Drifting is an unavoidable aspect of the MEMS gyroscopes used in this kind of device and without a magnetometer or positional tracking it cannot be compensated for exactly.

The good news is the amount of drift is generally consistent at a given temperature. By leaving the device still for a period of time, EDTracker monitors how much the drift is and then adjusts for this automatically.

If you skip this step, you may find your device appears to drift off to the left or right over time.

You can repeat this step as many times as you like. You may need to repeat it in the future, particularly if your environment changes (for example, you experience a very hot or very cold day compared to normal).

Performing the compensation calculation

First, let the device warm up. You can monitor the temperature of the device by looking at the red line on the graph at the bottom. When it has flattened off, your device is ready.

NOTE !	**During the following steps it is VITAL that the device remains completely flat and still.** Clamping the device with a suitable tool, taping it to or holding it under some books are all good suggestions. Failure to hold the device flat and still will result in poor calibration, and poor in-game performance.

Click ‘Reset View/Drift Tracking’ and observe the Yaw Drift value. Notice that a timer starts as well to provide details of how long the measurement has been running.

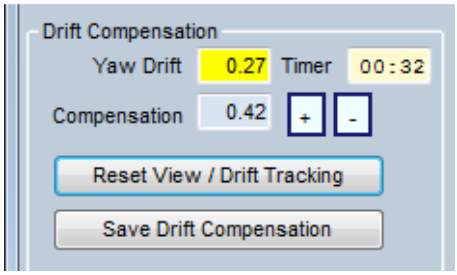


Figure 11- Reset View/Drift Tracking

1. Leave the device alone for 10 minutes or more. Put the kettle on, walk the dog, read the paper, entertain the kids. When you return, you should see some numbers specified next to the “Yaw Drift” value have averaged out.
2. Press the SAVE DRIFT COMPENSATION button, which will update the Drift Compensation value to the current yaw drift.
3. Leave the device for 30 seconds or so. If the Yaw Drift value is less than -0.1 or more than +0.1 then repeat steps 2 and 3. When it approaches zero (i.e. is between -0.1 and +0.1) you are all finished.

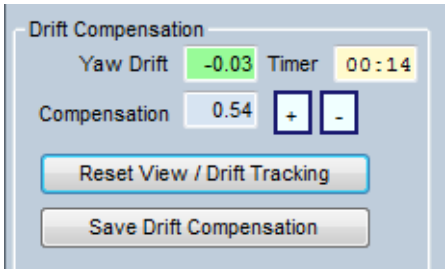


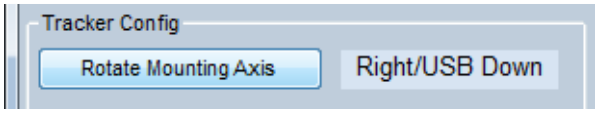
Figure 12- Green is Good

9. Configuring Preferences

The following actions can be performed via the GUI at any time. You can change these at will in accordance with your preferences.

Device Orientation

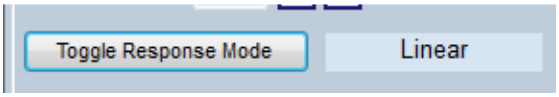
You can position the device on the top of your head, or on the sides. You need to set the orientation in the GUI to coincide with the mounting. By clicking the ROTATE MOUNTING AXIS button, you can cycle between various descriptions of the device position. Check the movement of the device with the movement 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.

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.

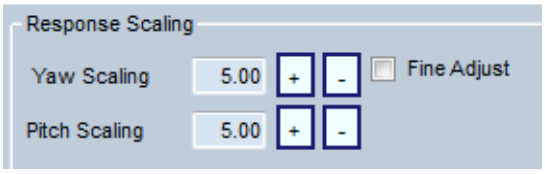


Exponential mode gives reduced responsiveness around the straight-ahead position, but the head movement then ramps up as you move off-centre. Linear is a more conventional mode where the responsiveness remains the same across the whole range of movement.

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.

Re-centre Hot Key

You can immediately re-centre the view in-game by pressing the button on top of the EDTracker. Alternatively you can assign a Function Key to perform the re-centre, e.g. F9

10. Using in-game

Joystick

The 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. Cocking 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.

Auto-centering

The EDTracker MPU-6050 software has a small degree of "auto centering" built in. If you look straight ahead and keep still for a moment, the EDTracker will adjust for any small drift it detects. You can see this in action by looking ahead after pressing the reset button and moving the view *very slightly* to the left or right. It should 'spring' back. Over the course of gameplay this spring-back helps average out any drift that may accumulate over time, but it does mean very small head inputs just off-centre should be avoided for prolonged periods (in reality, you very rarely do this anyway).

Generally it is advised to keep auto centering enabled to one of the preferred strengths (start with medium). Disable it for use with other head-tracking software such as Opentrack, but be aware that when disabled drift will become more pronounced if your calibration is not perfect.

Auto-centering can be disabled and enabled via the GUI :-

NOTE !	**The UI must be minimised for Auto-centering to function.** Auto-centering is disabled if the UI is not minimised. The UI will also consume some CPU if not minimised. Minimising the UI will send it to the Icon tray where it can be retrieved by double clicking on it.

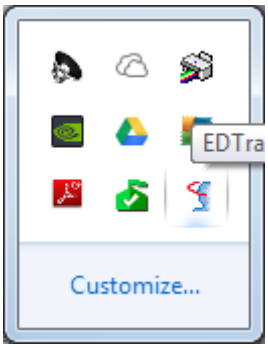


Figure 13 - EDTracker UI when minimized

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 a brief press of the button on the device. By pressing it, the device “resets” so that however the head tracker is currently positioned is what will be reported as “dead ahead”. You can perform this basic calibration as many times as you like – typically before starting your game. With the device in position on your head, look straight ahead at your screen (ie. Hold your head in the “straight ahead” position), press the button briefly and hold your head still for 2 seconds. The device will re-calibrate all axes to zero at this position.

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 calibration or you have configured the drift correction to an incorrect value. Re-tread your steps in the previous section slowly and verify you’ve done it right.

Elite Dangerous Beta

Some tips for those using the EDTracker with Elite Dangerous Beta v1.00+

Some tips for those using the EDTracker with Elite: Dangerous

- Set HEADLOOK MODE to DIRECT (not ACCUMULATE).
- Bind your up/down head motion to LOOK UP & DOWN AXIS
- Bind your left/right head motion to LOOK LEFT & RIGHT AXIS
- Do NOT bind anything to the LOOK UP, LOOK DOWN, LOOK LEFT or LOOK RIGHT controls!
- By default, headlook is turned off when you enter game. There is an option for DEFAULT HEADLOOK STATE – you can set this to ON if you prefer.
- You need to bind the “TOGGLE HEAD LOOK” button in your controls to a key or button on your joystick. Set it to TOGGLE ON if you don’t want to hold it down all the time!

Other Games

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