



# **SynDash System Specification**

**Version 1.6**

**Prepared by Sam Chen**

**RB-M1807, AiQ Synertial**

**01/25/2019**

## Table of Contents

Revision History	2
1. Introduction	3
About/Product Scope	3
Intended User	3
2. Overall Description	4
Product perspective	4
Comparison of the products	4
User Documentation	5
3. System User Interface	6
Live mode	6
Editor Mode	7
Editor Mode / Skeleton Properties	8
Editor Mode / Skeleton Properties / Flags	9
Editor Mode / Bone Properties	10
Editor Mode / Bone Properties / Sole-Segment	10
Live device	12
Recording playback	12
Hardware interface	13
Software interface	14
Communication interface	14
4. Other Requirements	14
5. Cobra 4G Calibration	15
1 How does a Cobra Glove or Mocapsuit calibrate?	15
2 Some example calibration poses:	18
3 ‘Squareness’ of the arc of calibration is of utmost importance:	21

## Revision History

Name	Date	Reason For Changes	Version
SynDash V1.5	02/08/18	Initial AiQ version with the following new functionalities: <ol style="list-style-type: none"> <li>1. Feature to config HUB through USB</li> <li>2. Sync module status notification</li> <li>3. Feature to automatically turn on “use sync” for HUB</li> <li>4. Vive trackers selectable and highlighted</li> <li>5. Vive trackers can be hidden if necessary</li> <li>6. Battery status for each suit</li> <li>7. Various enhancements and bug fixes</li> </ol>	1.5
SynDash V1.6	01/06/19	Current SynDash version with the following new functionalities: <ol style="list-style-type: none"> <li>1. Graphing for real-time and recorded data</li> <li>2. Various enhancements and bug fixes</li> </ol>	1.6

# 1. Introduction

## About/Product Scope

**SynDash V1.6** is the official Synertial tool to operate all of its 4th generation hardware products. It comes in multiple versions:

**SynDash Light** version has the following features available:

- Connects to the motion capture hardware (gloves + suits)
- Calibrates motion capture system
- Shows motions in real time
- Loads different skeleton files for the subject
- Records and re-plays the recordings
- Exports recordings to BVH

**SynDash Pro** is equipped with advanced features, including the skeleton editor system which allows additional fine tuning of AutoCal's Automatic Skeleton Extraction System, as well as sport graph mode.

## Intended User

**SynDash** is a system perfect for capturing every move of an athlete in action. The data can be analyzed in graph modes and avatar replays for the trainer or coach, with ability to compare limb performance in relation to other limbs or the ground.

## 2. Overall Description

### Product perspective

For over 25 years, Synertial has been doing kinematic research on sports analysis/training and human ergonomics. As pioneers of Inertial Measurement Unit (IMU)-based motion capture, we are well-aware how essential it is to start with accurate measurements of the subject’s skeleton.

AiQ-Synertial is the only motion capture company that offers an automated skeleton parameter extraction system (AutoCal) as well as the optical system hybrid system on top of that which allows, most importantly, the validation (and fine tuning if necessary) of the skeleton file exported by this type of extraction tool.

Once a skeleton is created and validated optically, the data produced by them is also validated and useful when compared with data produced by other mocap systems as well.

### Comparison of the products

	SynShow	SynDash-Light	SynDash-Pro
Shows real time skeleton model of the user	Yes	Yes	Yes
Speed of Capture (frames/second)	Based on HW	Based on HW	Based on HW
Captures Data	Yes	Yes	Yes
Shows, calibrates & captures multiple devices in the same scene	No	Yes	Yes
One keystroke, 2 poses calibration	Yes	Yes	Yes
Exports Data In Rotational BVH format	Yes	Yes	Yes
Exports Data In Joint Positions format	Yes	Yes	Yes
Exports Data in FBX	Yes	Yes	Yes
Replays or loads for review	Yes	Yes	Yes
Replay time scroll bar	No	Yes	Yes
Skeleton Loader	Yes	Yes	Yes
Skeleton Editor	No	No	Yes
Calibration Editor	No	Yes	Yes
Sensor number report	Yes	No	No
Configures Hardware	Yes	No	No
Live sensors state table	Text	Colored	Colored

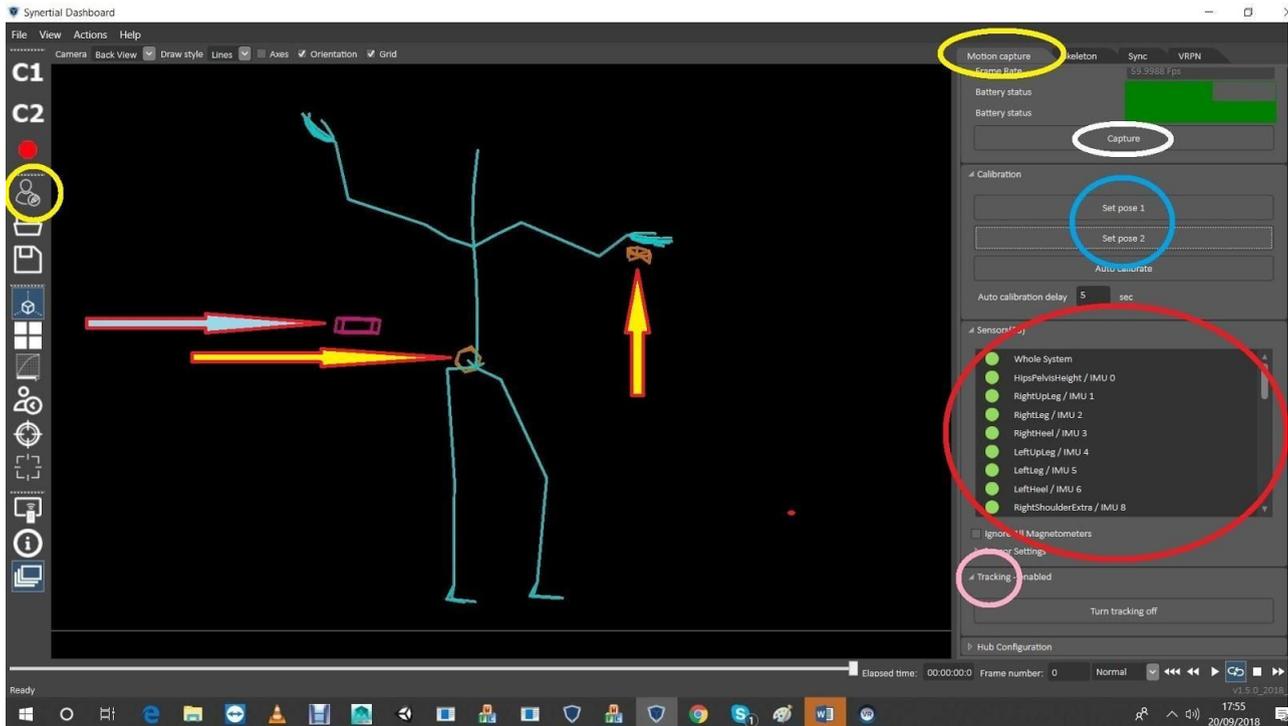
Records dashboard for random access to takes	<b>No</b>	Yes	Yes
Follows subject with replay/live camera	Yes	Yes	Yes
Real-time graphs	<b>No</b>	<b>No</b>	Yes
Max view windows	1	4	4
Battery level indicator	Text	Colored	Colored
Replays frame by frame	<b>No</b>	Yes	Yes
<b>Graph modes</b>			
Segment Rotations	N/A	N/A	Yes
Joint position relative to another position	N/A	N/A	Yes
Joint position absolute	N/A	N/A	Yes
Max graph view windows	N/A	N/A	Yes

## User Documentation

At the completion of development, a user manual and implementation guide will be made available online to the customer.

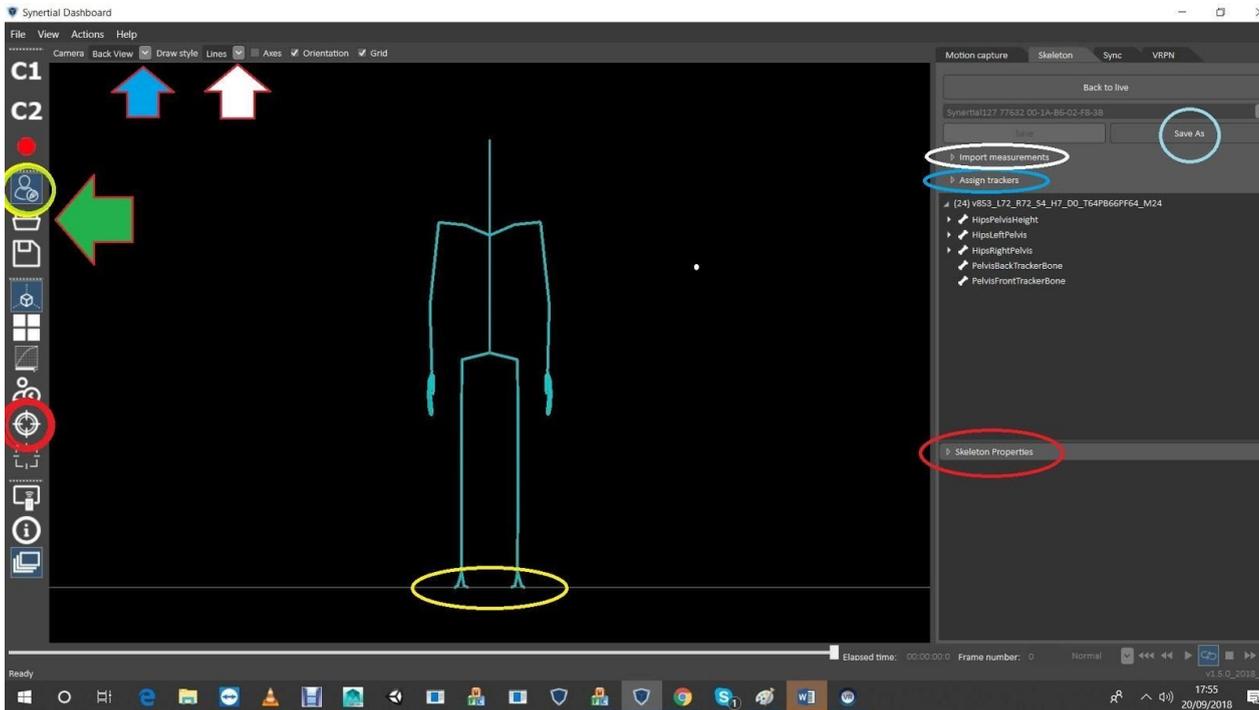
### 3. System User Interface

#### Live mode



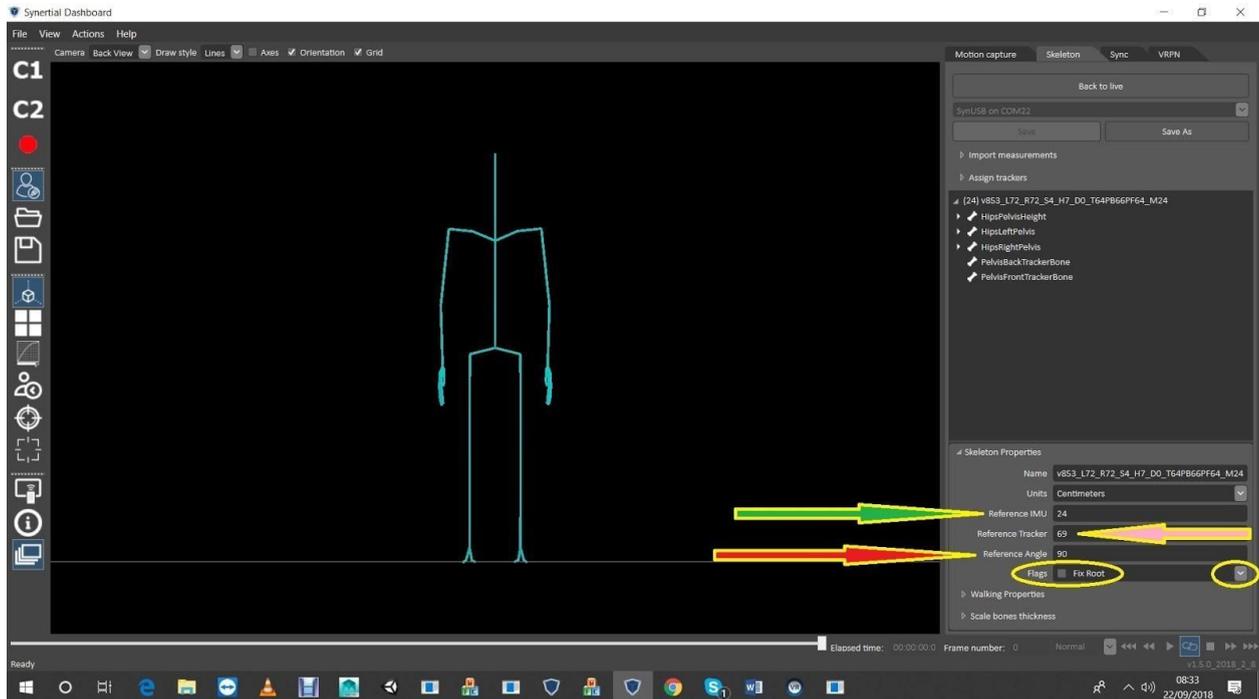
- Yellow circle: Toggle button to go between 'Live' and 'Edit' modes
- Yellow arrows: Vive pucks on coccyx and left hand
- Blue arrow: Vive 'Controller'
- Yellow oval: 'Motion Capture' tab manages the HW and the 'Live' mode
- White oval: Start and Stop capturing the live feed
- Blue circle: Pose-1/Pose-2 calibration commands
- Red circle: the 'State' of sensors (green, yellow, red, grey: good to go, almost good, not good, and 'unreliable'). Every sensor has its own reporting, including the Vive sensors (scroll to the bottom)
- Pink circle: 'Enable' or 'Disable' the Vive live data

## Editor Mode



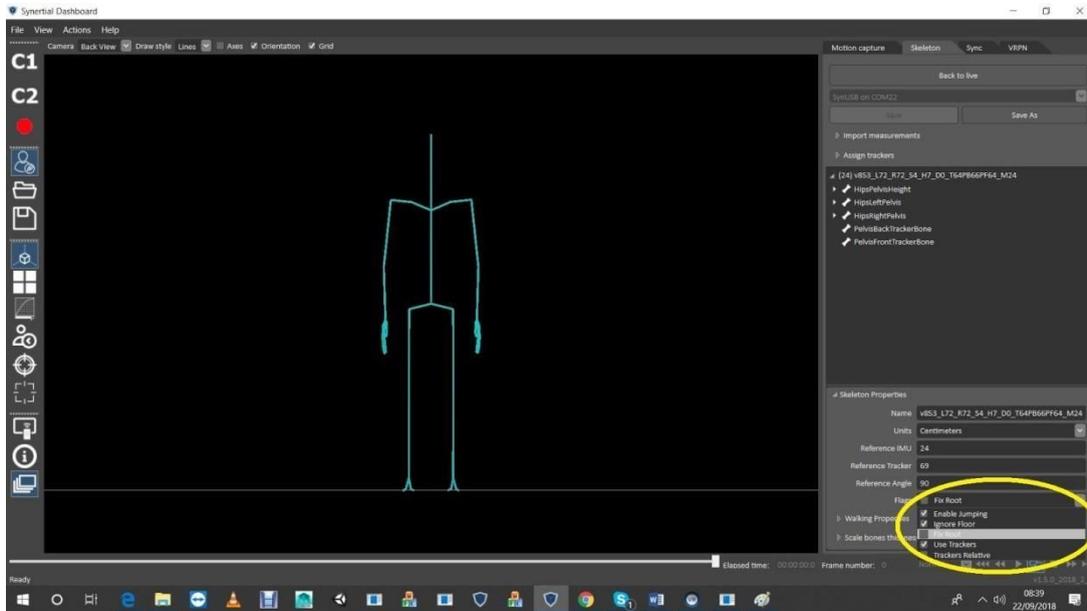
- Yellow circle: Toggle button to go between 'Live' and 'Edit' modes.
- Green arrow: Load SK (skeleton) file
- Blue arrow: Viewer's modes (for this exercise) are 'Left Side' and 'Back View'
- White arrow: Thickness of bones
- Yellow oval: When in 'Edit' mode, feet will always be on the ground. In 'Live' mode, the puck dictates how far off the ground the feet are
- Blue circle: Save the skeleton often
- White oval: Import Kinexact or AutoCal (hand & body) data
- Blue oval: 'Assign Tracker' button to change the ID number on Vive puck and controllers
- Red oval: Click the dropdown to see 'Skeleton Properties'

## Editor Mode / Skeleton Properties



- Green arrow: 'Reference IMU' is the sensor used by Pose-1/Pose-2 process to calibrate. Find sensor numbers using SynShow if not certain of the sensor numbers.
- Pink arrow: 'Reference Tracker' is the Vive Tracker used to fuse the root of the suit/skeleton into the Vive space (through the same Pose-1/Pose-2 procedure used to calibrate the suit with the 'reference IMU', simultaneously)
- Red arrow: 'Reference Angle' is the axis and direction of Pose-1/Pose-2 angular transition:
  - 0 & 180 are in 'X' (in either direction)
  - 90 & 270 are in 'Y'
- Yellow ovals: Drop-down the 'Flags' to define the relationship between the suit root (and its walking method which uses the Lowest Point Algorithm, LPA) and trackers for the global coordinates: how much of each? Or if one without the other?

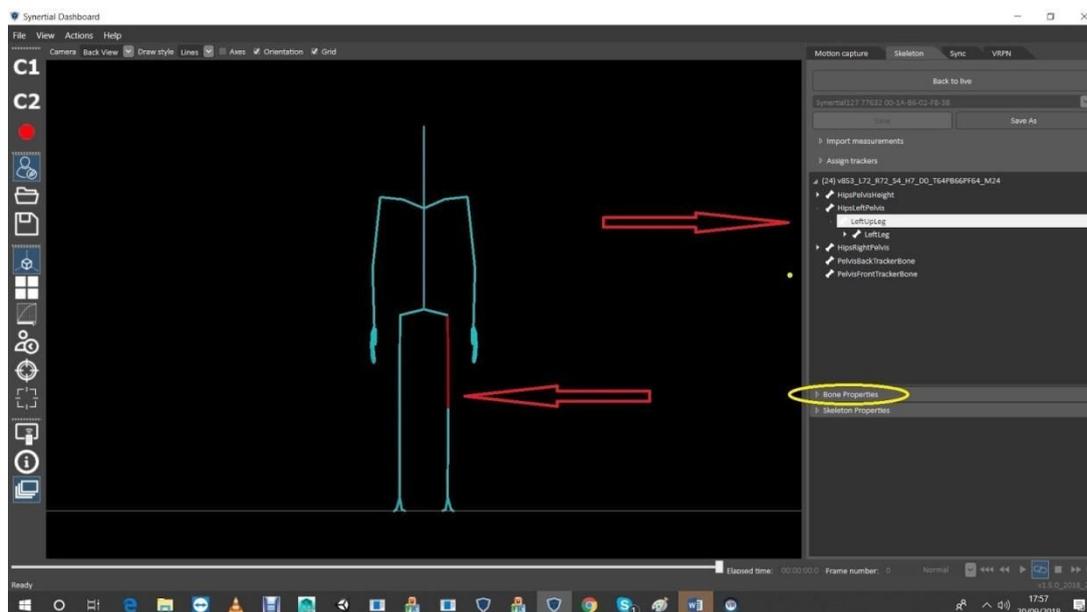
## Editor Mode / Skeleton Properties / Flags



- In this exercise, the trackers need to be 100% and no LPA throughout the process.

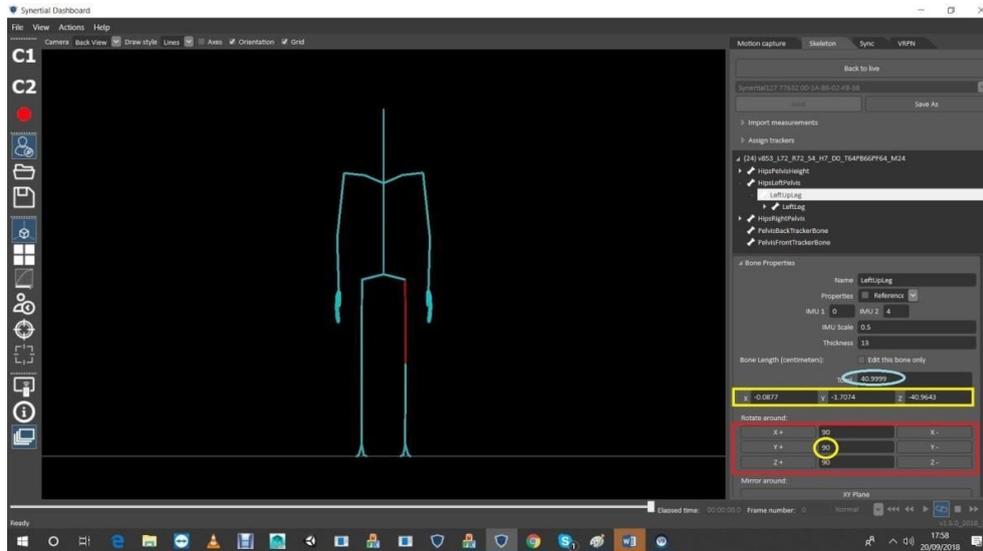
**TO EDIT SKELETONS:** modify the **length** or **orientation** of any bone, highlight it in the **Editor Mode** or click on the bone stick figure.

**Selecting a bone to modify:**



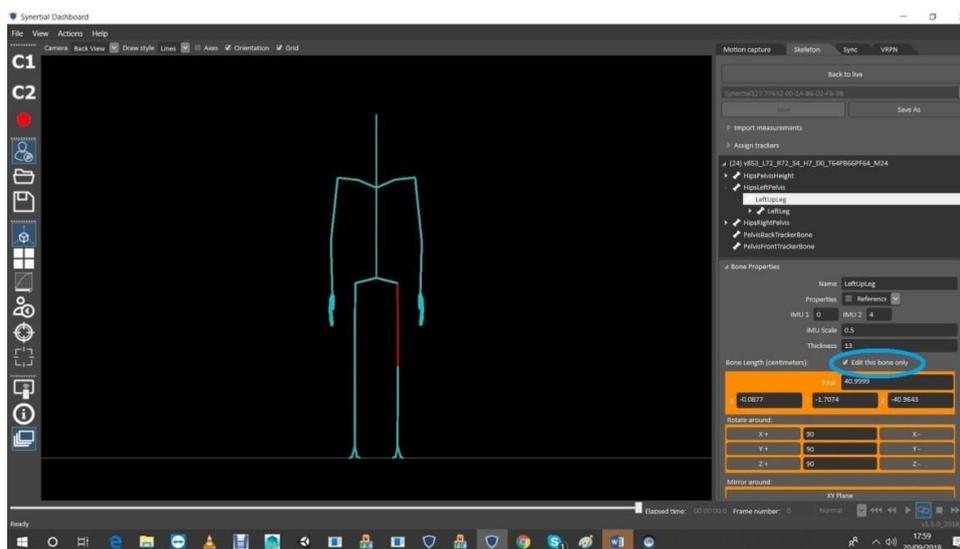
- Red arrows: Highlight a bone by directly double clicking on the stick figure or use the tree on the upper right (triangles on the left of titles can further expand the tree)..
- Yellow oval: Drop down to modify the particular bone (segment)

## Editor Mode / Bone Properties



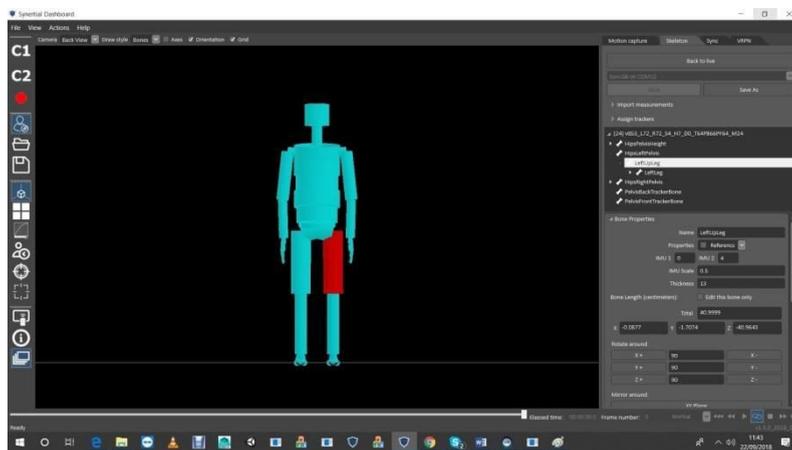
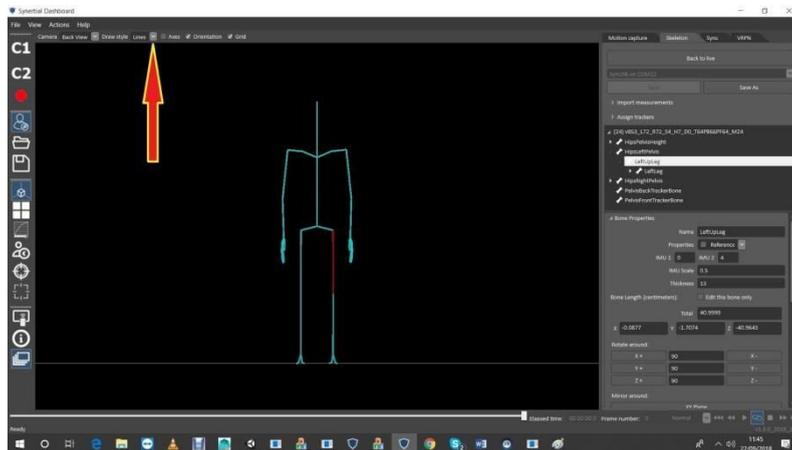
- Blue oval: the 'Total' length of that segment, from its 'Parent Node' to its 'Child Node'
- Yellow rectangle: Distance of a Child Node from its Parent Node in each global axes
- Red rectangle: Where the orientation of the blue oval ('Total' above) can be modified
- Yellow circle: Increment in degrees by which clicks on either side rotate the segment. Modifying the orientation of the segment using the red rectangle will change the values in the yellow rectangle automatically.

## Editor Mode / Bone Properties / Sole-Segment

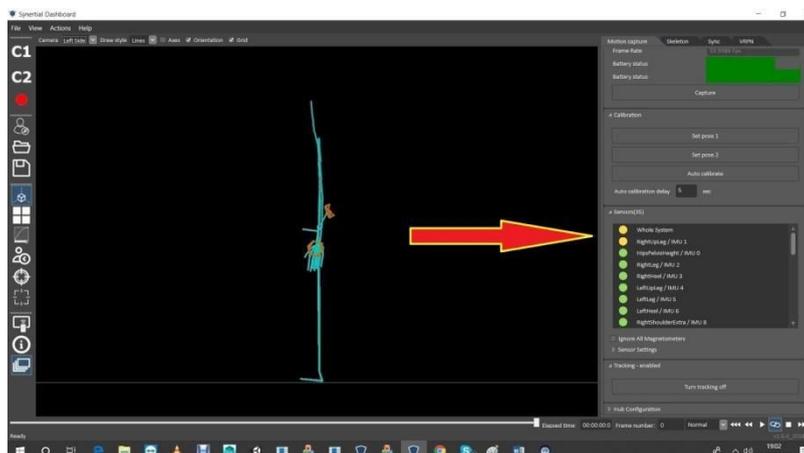


- Blue oval: When 'Edit this bone only' feature is checked, segment edits are contained and will not alter the parameters of the Child segments thereafter. This is useful for example, when you want to only modify the length of the upper leg without changing the distance between the root and floor.

## Change Draw style from Bones to Lines and back

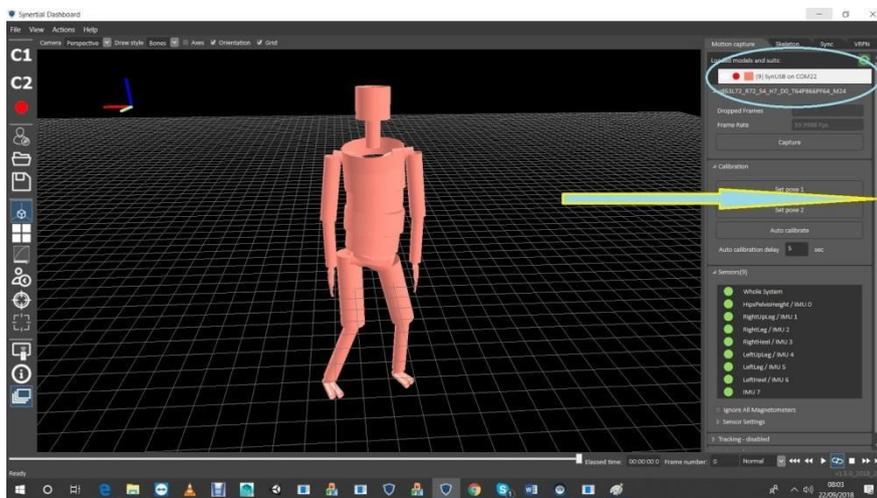


OK when sensor 'State' lights go yellow during this process



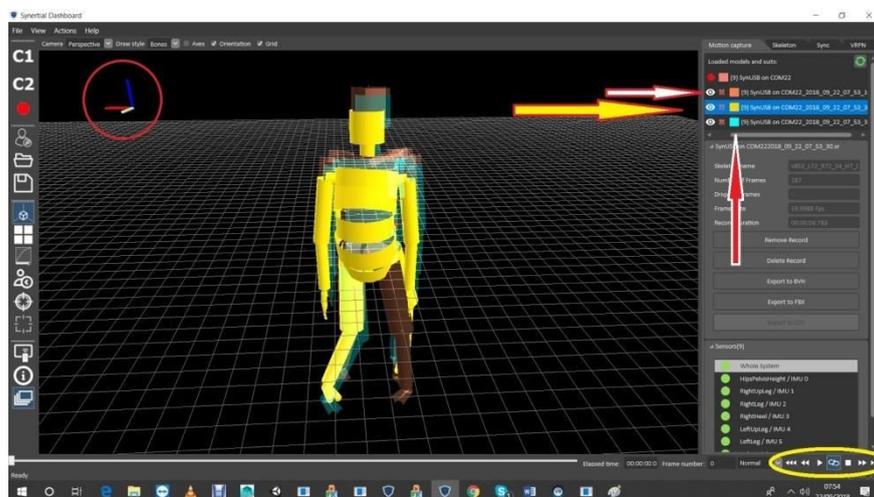
- The yellow circles indicate which sensors (segments) will 'stay' uncalibrated after pose-1/pose-2 refresh. This is OK during this exercise.

## Commands for capture and replaying recordings Live device



- Blue arrow shows the scroll bar for the whole right column.
- Blue oval shows there is one 'live' device present,
- Change the color by left clicking on the square on the left of the name,
- Click the red circle to start recording and the new button appearing then to stop recording
- To the left of the red recording button there is an 'eye' icon that will make things visible or not

## Recording playback



- Arrows showing 3 separate recordings (right click on the name to see where the data is automatically stored)
- Yellow oval showing the playback controls (for the highlighted recording)
- Can change color and visibility of each recorded skeleton

## Hardware interface

The next generation, G5, of AiQ Synertial motion capture suit is still under development. But here is an example of the current Synertial suit:

- First-ever wearable sensor suit with no root drift
- Everything measured, no guesstimations
- Best-in-Class Cobra Gloves for fingers (7, 13, 16)
- Slap-on the Wireless Vive 'Puck' anywhere
- One keystroke to calibrate Vive & suit together
- Use additional 'Pucks' for robust sitting/reclining
- Use trackers to refine subject's base skeleton
- Walk in and out of the Vive space of 40 m<sup>2</sup>
- Restart HW & WiFi in 3 seconds
- Sockets and sensors included for props
- Include other Vive trackers in Synertial captures
- Up to 55 IMUs @ 240fps, 10 trackers, & sync HW



Pose 1

Pose 2

**Software interface**

- MotionBuilder plug-in
- Unity plug-in
- Unreal plug-in

**Communication interface****Wifi:**

- Up to 240 HW fps; 120 standard
- AA Batteries, 5vDC wall adapter or power bank
- Separate sync HW up to 1000 FPS (Standard 29.97 FPS), drawing power from main hub
- Access-point agnostic WIFI
- Multiple PCs can run the same HW
- 12.8cm x 7.8cm x 2cm with locking connectors

**Bluetooth:**

- Specification TBD

## 4. Other Requirements

**System Requirements:**

- Windows 10
- VR-Ready system with at least an NVIDIA GTX-1060
- i-7 processor and at least 16Gs of RAM
- Vive-Pro System for the Opto-inertial Hybrid Mode
- 4th Gen or 5th Gen AiQ-Synertial devices (suits or gloves)

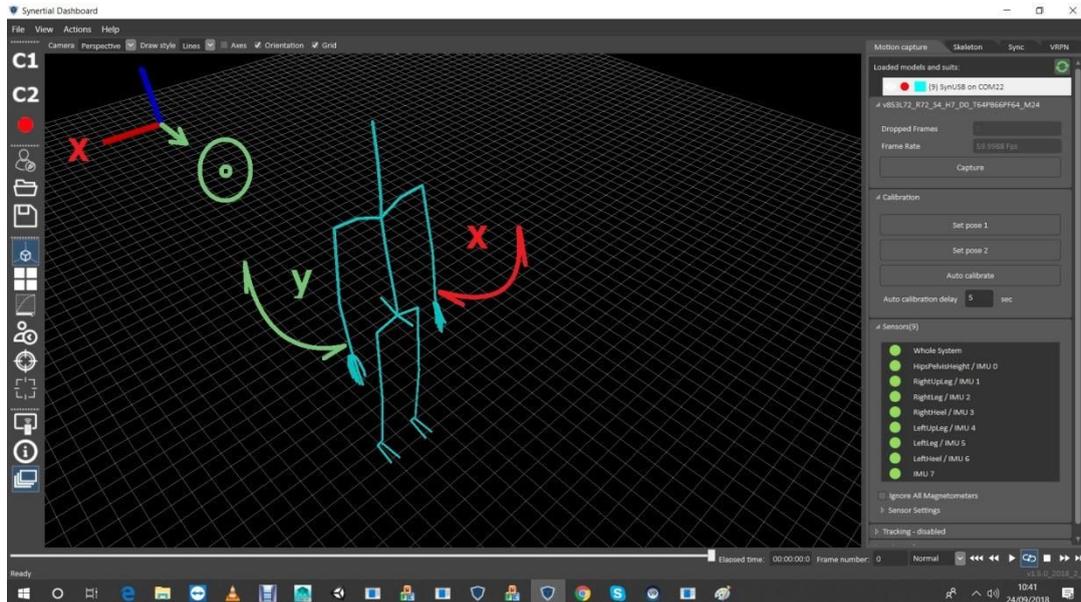
Safety requirements: N/A

Security requirements: N/A

Software quality attributes: N/A

## 5. Cobra 4G Calibration

Cobra template skeleton files will start off looking away, into y+ direction.

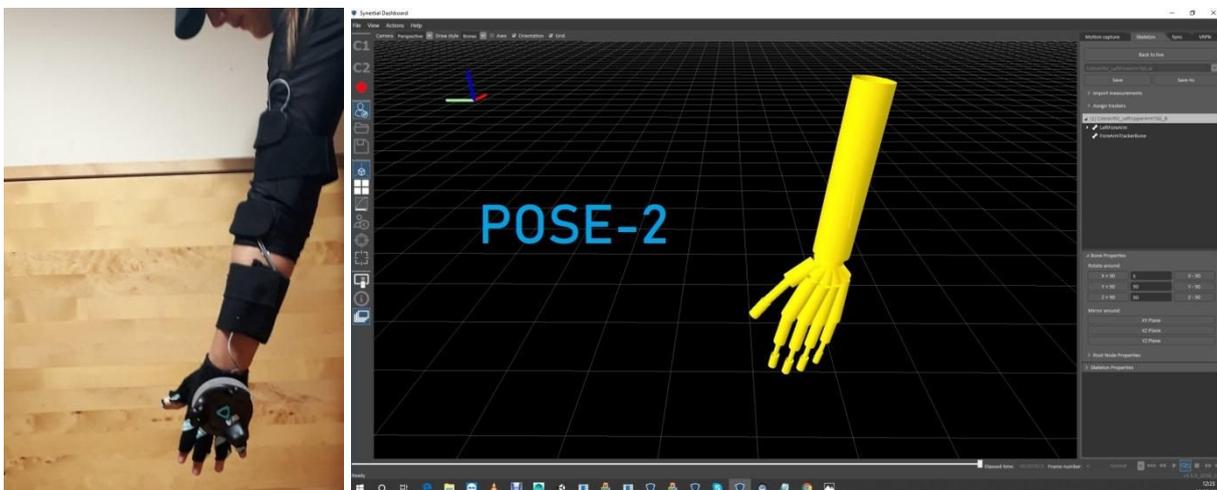


# 1

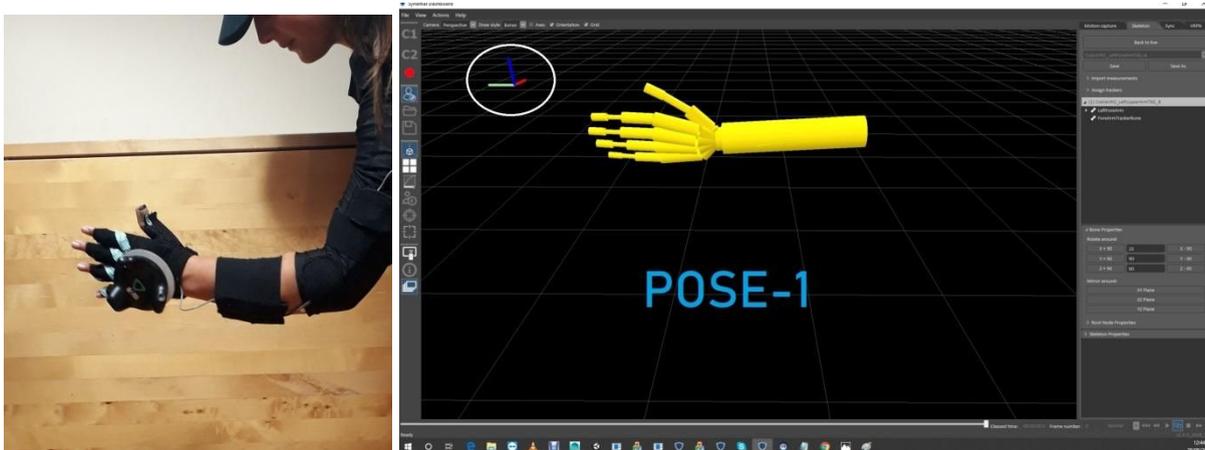
**How does a Cobra Glove or Mocapsuit calibrate?**

2 poses need to be sampled (a keystroke trigger) to complete a calibration:

**2nd pose** (pose-2) this is dictated by the skeleton file. It's the first frame of the rest of this live stream. Rosie should also have had her fingers closed to look more like the pose-2 structure showing when pose-2 is triggered (her fingers will stay that much apart thereafter).



**First pose** (pose-1) is a resting posture from which going into pose-2 can be completed by rotating at least one sensor, and around **only one** axis (in this case around the 'X', note the coordinates inside the white circle: **Green Y forward, Red X to the right and Z up**).

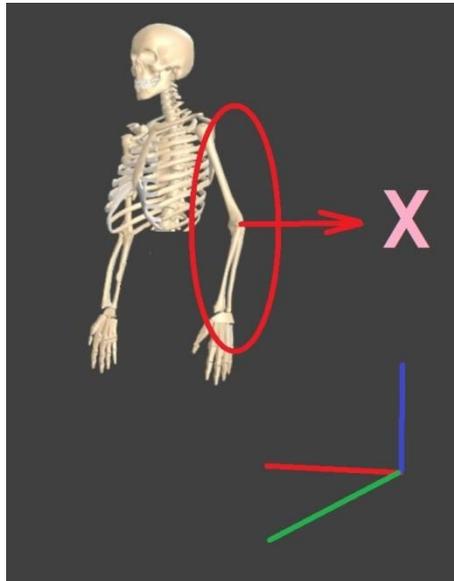


**Expanding:** In the above example, Rosie configured the **calibration parameters** like so:

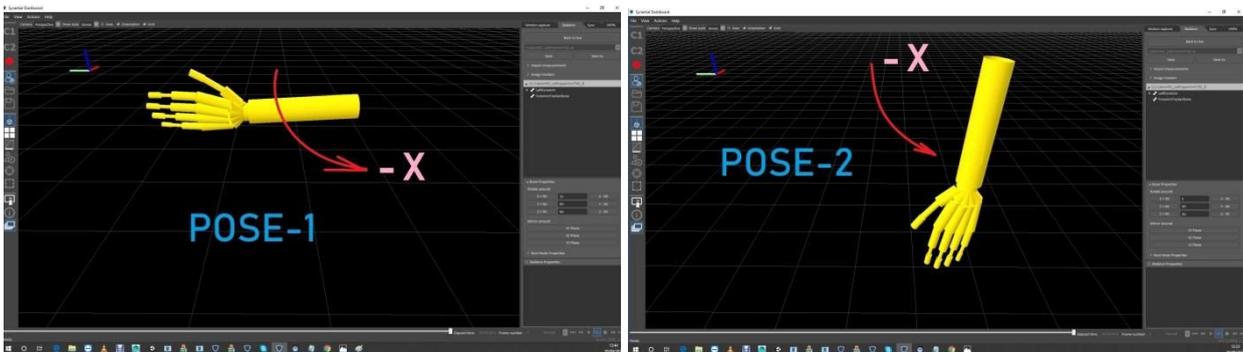
**1.1 Sensor number 24 was selected** (Reference IMU) which is the palm sensor here (she made sure it was 24 by going to SynShow and 'ctrl+R' till the live sensors come on, and wiggled her palm sensor and made sure number 24 on the screen corresponded satisfactorily).



**1.2** X was selected as the axis (Reference Angle) so sensor 24 needs to rotate around a *clean, upright arc in X axis*. In this example rotating the elbow provides the cleanest arc.



**1.3** Minus direction from Pose-1 to Pose-2 (Also part of Reference Angle).



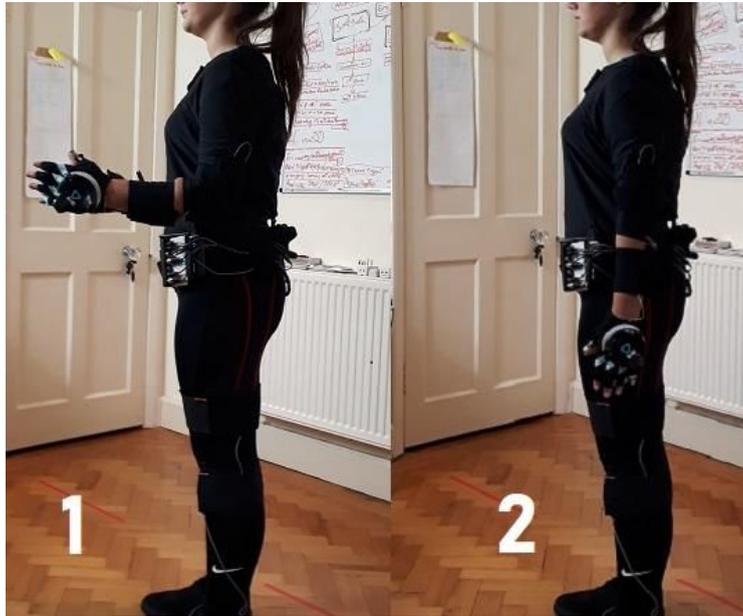
**1.4** Hold it in pose-1, trigger pose-1 on the keyboard, then hold pose-2 and trigger again.

-----  
 Notes: In the skeleton file (sk file), 0 & 180 are the degrees invoking -X and +X, and 90 & 270 for -Y & +Y. Z has not been used or tested much, but possible 45 & 135.  
 -----

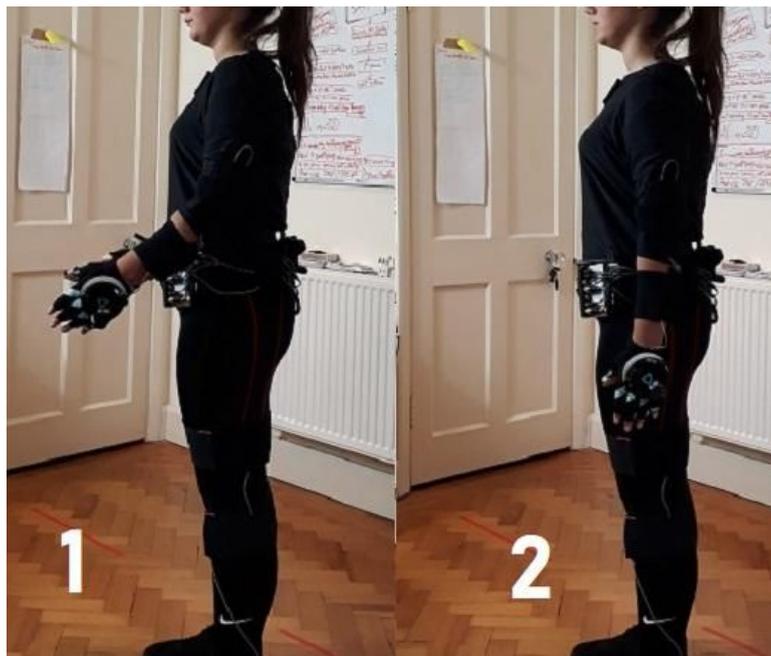
# 2

Some example calibration poses:

A- This one can be both for glove or suit (or gloves with suit)



B- This one brings the exact outcome as example A (above) does



C- This one is in Y axis for suit, gloves or suit with gloves



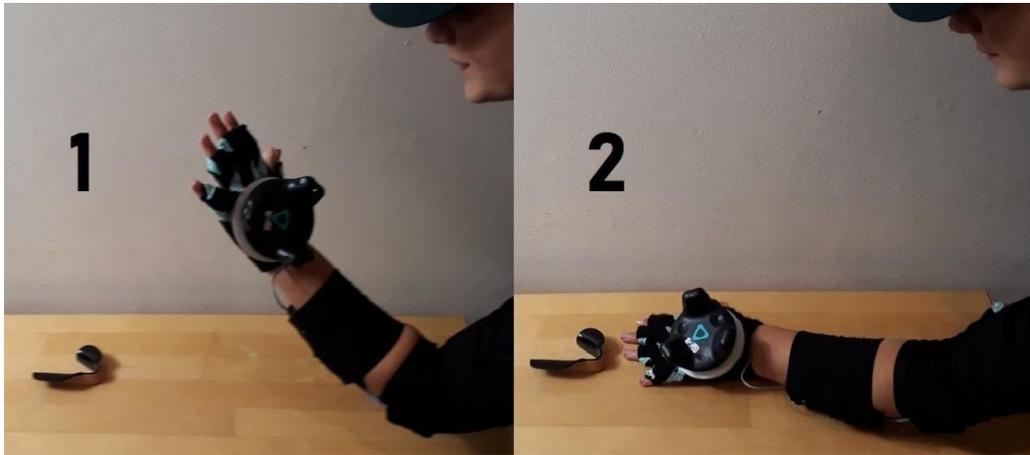
D- This one is the palm on the table pose-2, rotate elbow in X, and using the 'mini' jig #3 between the thumb and index finger. 20 to 30 degrees between poses 1 & 2.



NOTE: using jig #3 try to push it in all the way between thumb & index and press index against the side. Palm should be pressed down against the table top



E- Same as D above from a different angle of view and 'mini' jig #3 wasn't used.



F- This one uses jig #2 which can be used for suits, glove(s) and suit with gloves.



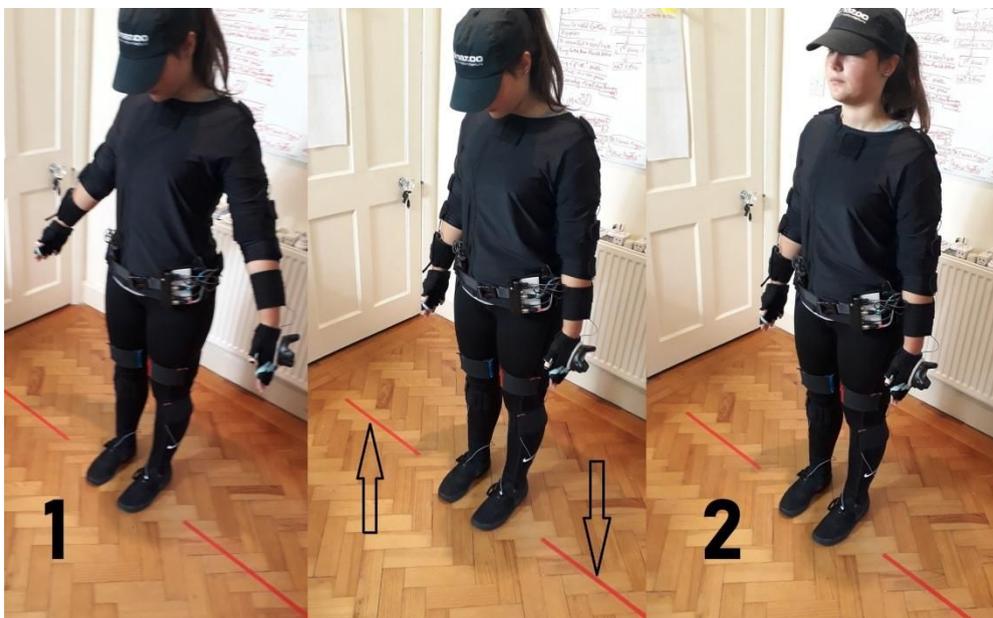
# 3

‘Squareness’ of the arc of calibration is of utmost importance:

Below, Ami is trying to square her swing between poses



Below, Rosie is keeping rotation in Y square using tape lines on the ground



- She triggers pose-1, making sure she's on the line, looking down from corners of her eyes;
- Then she bring her arms to the side while still keeping an eye on the lines;
- Then she looks up;
- Then she triggers Pose-2