

# 光学传感器校准

## SteamVR™ Tracking

### 简介

在开发 SteamVR™ 定位对象时，系统确切地了解光学传感器在定位对象上所处位置至关重要。尽管根据机械模型的指示，JSON 文件可能配置有传感器的位置，但在装配过程中产生的亚毫米异常都会给对象的定位性能带来不良影响。要抵消这些异常，应在装配后对每个对象执行简要的校准程序。此程序会产生一个新的 JSON 配置，使传感器更接近于其实际位置。

### 概述

定位对象的校准例程需要与正常操作使用的相同硬件。将一份理想的 JSON 配置文件提供给校准工具，然后在 VR 空间内移动和转动对象。校准工具会记录传感器命中情况，并尝试拟合出一套与所记录的数据最匹配的传感器位置和方位。这些拟合效果最佳的位置和方位将通过一个新的 JSON 文件提供，该文件可编程到对象中以提高性能。

### 设定

要执行校准，您需要用到：

1. **vrtrackingcalib.exe** - 此命令行实用程序位于以下目录中：

```
C:\Program Files (x86)\Steam\steamapps\common\SteamVR Tracking HDK\tools\bin\win32
```

2. **定位对象** - 完全组装且能够进行基本定位的对象。如果您的对象无法引导到 SteamVR™ 环境，您将无法对其进行校准。
3. **JSON 文件** - 此文件应是从对象的 3D 模型直接生成的一个理想的 JSON 文件。请注意“device\_serial\_number”字段必须与您要尝试校准的对象的序列号一致。这是校准工具识别对象的方式。请勿使用上次校准例程时所生成的 JSON 文件。在后面校准中使用之前的校准结果会导致小错误累积起来，类似于对复印件进行复印。
4. **定位器** - 单个定位器。如果您有一个配有两个定位器的标准 VR 系统，关闭其中一个定位器执行校准程序。
5. **空间** - 当您能够在空间内移动可定位对象时，校准例程将达到最佳效果。理想条件为 2 x 1.5 米的标准 VR 空间。

# 运行校准

校准工具具有很多功能超出本文档范围的功能。下列程序介绍了标准机体校准。

vrtrackingcalib.exe 是一个命令行实用程序，它必须从 Windows 命令提示符调用，并且提供有多个运行时间标志位，如下所示：

```
> vrtrackingcalib.exe /usedisambiguation framer /bodycal <your JSON file.json> 800 200
```

运行时间标志位如下：

- **/usedisambiguation** - 写入时，默认消歧工具不支持新光学传感器模型的输出，因此必须指定“framer”特定工具。
- **/bodycal** - 指明您想要执行传感器机体校准。下列三个参数是对象的 JSON 文件以及两个数值，这两个数值分别表示所需的传感器命中次数以及每个传感器每秒所需的命中次数。数值越多，收集的数据就越多，就会产生更好的结果，但也会需要更长的时间来完成校准。

在命令行中输入相应的标志后，按 Enter 键以运行校准例程。Vrtrackingcalib 会输出大量描述其设置程序的信息。最终，vrtrackingcalib 为校准准备就绪：

```
Ready to run capture position number 0.  
Press <enter> key to begin:
```

如果您没有看到此提示符，请确认对象已启动且在定位器的视野范围以内。还请确认对象已被引导至 SteamVR™ 环境。

按 Enter 键开始校准过程。此时，目标是在定位器有效区域内，将对象引入尽可能多的各种平移和旋转位置。工具会通过一行不断增长的点线来表示已记录的位置和方位的数量。如果对象保持静止，这些点和校准例程将暂停。您可能需要使用大量的物理空间以获得最佳结果。

最后，vrtrackingcalib.exe 输出一行类似下行的内容：

```
Sensor IDs TBD:  22 (0-hits) 24 (0-hits) 20 (0-hits) 18 (0-hits) 14 (0-hits) 16 (0-hits) 12 (0-hits) 10 (0-hits) 8 6 4 2 0 (0-hits) 30 (0-hits) 29 31 1 3 5 7 9 15 11 13 17 19 21 23 25 27 28 26 (0-hits)
```

这列出了所收到的命中次数尚未达到运行时间参数中所规定最低命中次数的传感器。它还专门列出了所收到的命中次数为 0 的传感器。校准期间的目标是为了确保对象上的每个传感器所收到的命中次数均达到所需的最低要求。如果传感器未获得足够的命中次数，旋转对象以便让传感器能够看到定位器。

最后，列出的传感器数量将减少：

```
Sensor IDs TBD:  16 10 8 6 4 0 1 3 5 11
```

当数量降低到零并且记录到运行时间参数规定的所需总命中次数时，校准例程将停止并显示结果。

## 校准结果

大多数结果数据与基本机体校准无关，但有几个重点。

```
Termination: CONVERGENCE
```

这说明校准例程成功执行，并且 vrtrackingcalib.exe 能够根据所收集到的数据拟合一份传感器布局图。如果您没有看到此结果，请确认您的传感器正在收集数据并且您的 JSON 配置文件已尽可能地对它们进行了映射。

```
Corrected scale of 0.9901
```

这说明了定位对象与其 JSON 配置文件相比大多少或小多少。越接近 1 越好。

```
Corrected normal shift of -0.0013
```

此数字表示实际光电二极管位置沿传感器平面法线方向相对于 CAD 模型中位置的偏移量。这通常是一个很小的数字，仅占到覆盖光电二极管的透镜的厚度。

```
Corrected rotation of 0.0583 deg
```

这说明了定位对象与其 JSON 文件相比的转动情况。越小越好。0 为最佳。

```
Corrected translation of 0.1459 mm
```

这说明了定位对象与其 JSON 文件相比的偏移程度。越小越好。0 为最佳。

如果您的任何结果远离理想值，请检查对象和 JSON 文件以查看是否存在任何重要错误。

除上述调试信息以外，工具还将生成两个 JSON 文件：calib\_observations\_XXXXX.json 文件（其主要包含与命令行输出相同的内容信息）和 auto\_<serial number>.json 文件（其包含与对象的原始序列号相同的信息，但还包含调整后的 modeNormal 和 modelPoint）。如下所示，新的值与初始值略有不同，其精确度更高。将新的 JSON 文件上传到对象上以提高定位性能。

#### JSON 校准前:

```
"modelNormals" : [  
  [0.851,0.397,-0.342],  
  [0.397131, 0.85165074, -0.34202014],  
  [0.707107, 0, -0.70710678],...
```

#### JSON 校准后:

```
"modelNormals" : [  
  [ 0.85152214765548706, 0.39724358916282654, -0.34220984578132629 ],  
  [ 0.3971310555934906, 0.85165083408355713, -0.342020183801651 ],  
  [ 0.70710694789886475, 0, -0.70710670948028564 ],...
```