



PERCEPTION NEURON UNITY 使用手册

PerceptionNeuronUnityIntegration 0.2.17

诺亦腾科技有限公司: <http://noitom.com.cn/>

08/31/2020

目录

1. 简介	3
2. UNITY 动捕资源包导入	4
3. PERCEPTIONNEURON.UNITYPACKAGE 资源包简介	5
3.1 示例场景	5
3.2 核心脚本	5
3.3 NEURONTRANSFORMINSTANCE.CS 和 NEURONANIMATORINSTANCE.CS 的区别	7
3.4 NEURONANIMATORINSTANCE 脚本中公共变量说明	7
4. 如何在 AXIS 软件中设置数据流转发	9
4.1 AXIS NEURON/AXIS NEURON PRO 软件设置	9
4.2 AXIS STUDIO 软件设置	9
5. NEURON SKELETON TOOLS	11
6. 如何配置/选择动捕模型	12
7. 如何使用 AXIS 数据流实时驱动 UNITY 中模型	14
8. 头显集成	18
8.1 头显与 AXIS 同步集成注意事项	18
8.2 示例脚本通过下列方式工作	18
9. 如何使用 AXISSTUDIO 导出的 FBX 文件驱动模型	19
9.1 HUMANOID 方式	19
9.2 GENERIC 方式	23
10. 附录 A: NOITOM 骨骼映射	24
10.1 PERCEPTIONNEURONSTUDIO 骨骼映射	24
10.2 PERCEPTIONNEURON 骨骼映射	25
11. 附录 B: 骨骼附录	26
12. 附录 C: 二进制数据序列	28

1. 简介

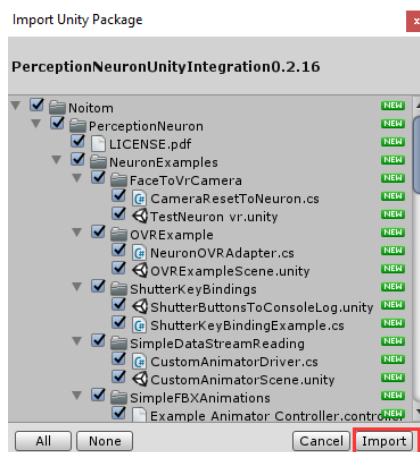
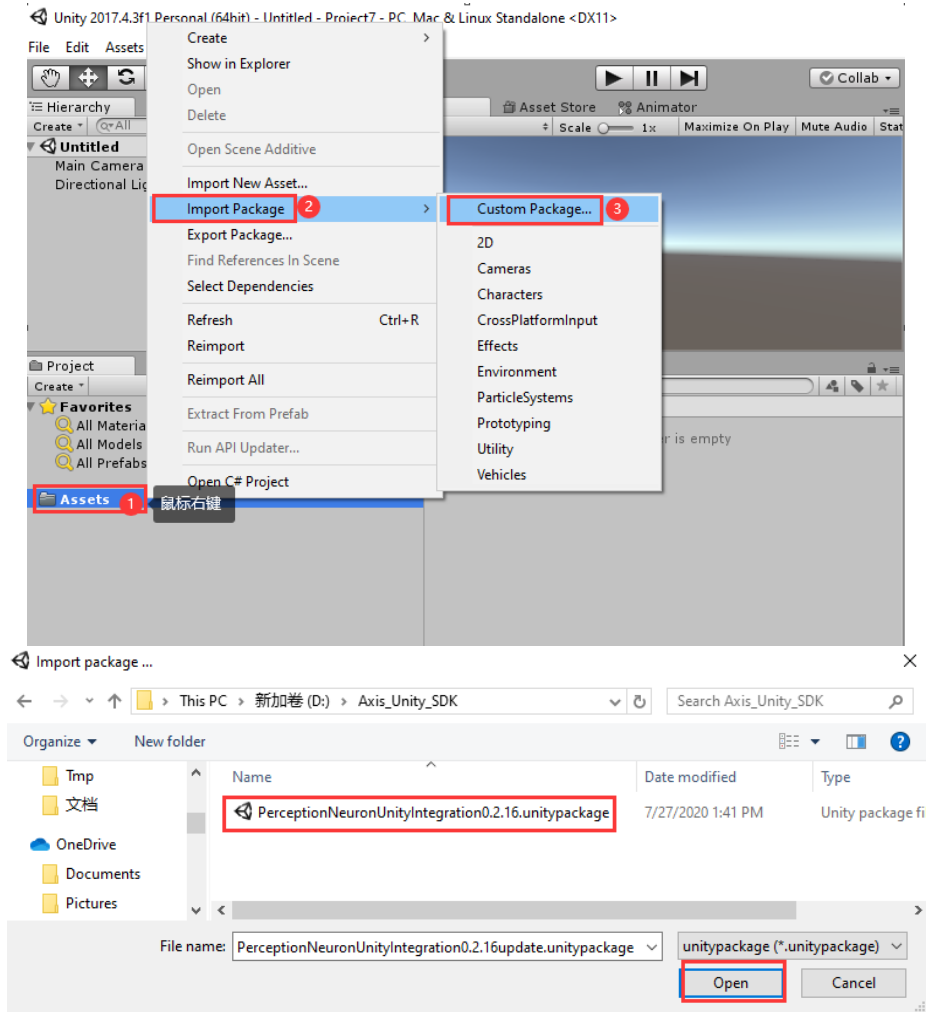
本文档将帮助你了解如何使用游戏引擎 Unity 3D 读取诺亦腾 Axis 系列动捕软件的运动捕捉数据。AxisStudio 软件支持实时传输离线数据和实时数据，离线数据指回放的动捕数据，实时数据是指连接设备时的数据，实时传输是指打开数据流开关后，AxisStudio 软件中的离线数据/实时数据会通过网络协议同步传输到 Unity 3D 软件中。为了方便测试我们可以直接使用已经录制好的数据来进行数据流测试。

数据流通过网络连接将 Axis 软件中角色的动画数据实时传输到 Unity 3D 软件中，所以只需要满足 Axis 软件与 Unity 软件能建立网络连接即可，不要求必须在同一台电脑上运行。

如果使用中
发现缺陷或者需要帮助
请通过该邮箱地址直接联系我们：Noitom_service@noitom.com

2. Unity 动捕资源包导入

打开 Unity 工程后点击 Asset > Import Package > Custom Package 将 PerceptionNeuron.unitypackage 添加到 Unity 工程中即可。

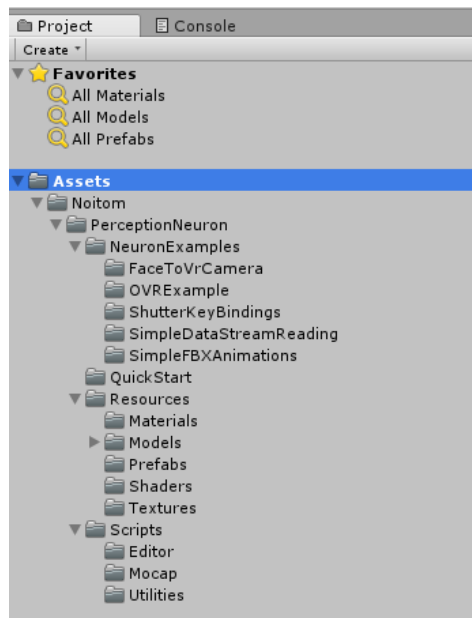


3. PerceptionNeuron.unitypackage 资源包简介

3.1 示例场景

在 Assets >Noitom >PerceptionNeuron >NeuronExamples 中预置了几个示例场景，简介如下：

1. FaceToVrCamera: 头显示例，vr 摄像机朝向和头朝向一致
2. OVRExample: 头显示例，头显跟着头结点
3. ShutterKeyBinding: Shutter 热键示例
4. SimpleFBXAnimation: FBX 动画文件应用示例



3.2 核心脚本

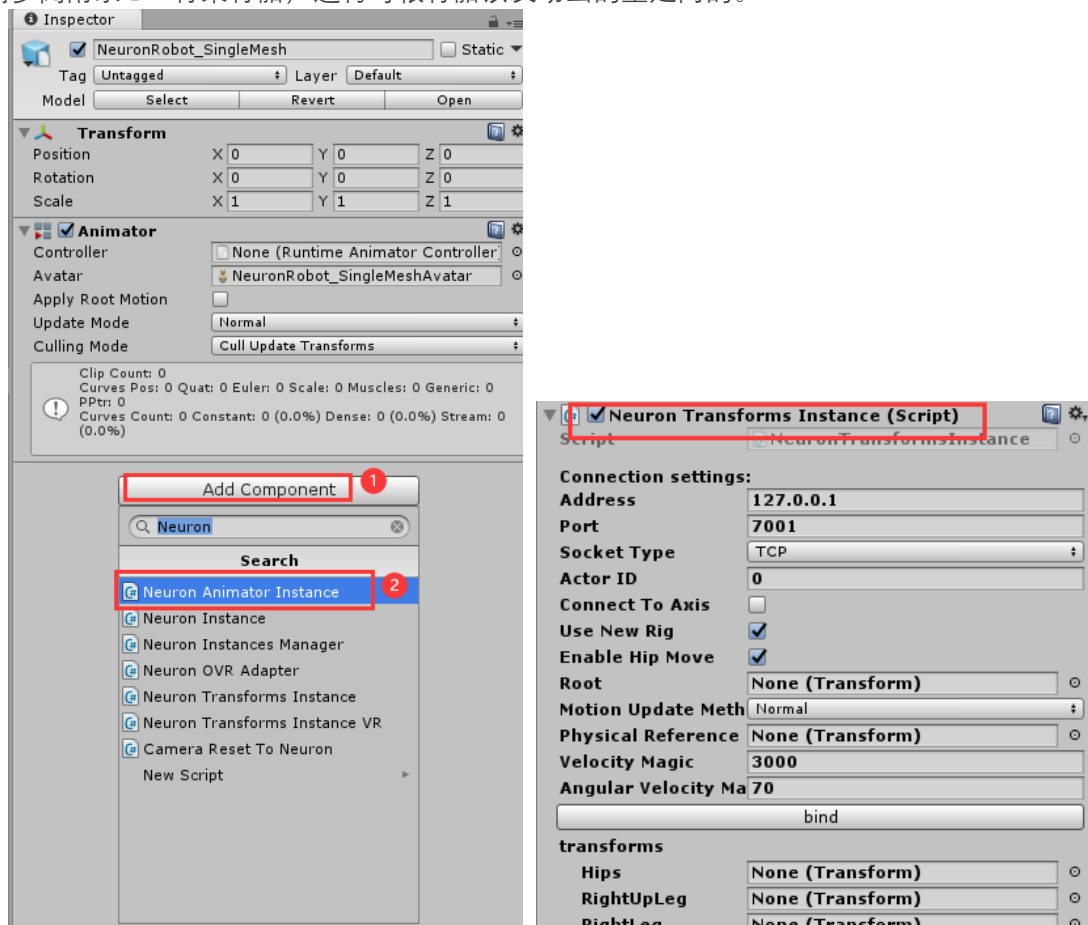
1. Assets/Neuron/Scripts/Mocap/NeuronDataReaderManaged.cs
管理接收来自 Axis 软件的 bvh 数据
2. Assets/Neuron/Scripts/Mocap/NeuronConnection.cs
管理与 Axis 软件的数据连接，支持多实例连接
3. Assets/Neuron/Scripts/Mocap/NeuronSource.cs
NeuronSource 通过 activeActors 和 suspendedActors 字典来管理 NeuronActor 的实例。
NeuronSource 使用 OnUpdate 方法监视 NeuronActor 内部的最新时间戳更新，并使用阈值来判断是否丢失了任何参与者（Axis 中的参与者数量已更改或连接完全丢失）。发生这种情况时，NeuronSource 将在两个词典之间添加或删除 actor，并通知 NeuronActor。

4. Assets/Neuron/Scripts/Mocap/NeuronActor.cs
仅存储最新运动数据帧的数据类，还提供了一些方法来解析从网络以浮点值形式接收的接收到的运动数据。 NeuronActor 还可以保存 Mocap 信息，并提供在 NeuronSource 恢复或暂停回调时注册回调的方法。
5. Assets/Neuron/Scripts/Mocap/NeuronInstance.cs
用于接收运动数据的各种实例的基类。继承自 UnityEngine.MonoBehaviour。 NeuronInstance 提供状态更改以及从 NeuronActor 实例接收 Mocap 信息的回调，该信息通过连接或其他方法绑定到此实例。此类不能直接使用，但可以继承以提供自定义方法来应用运动数据，处理状态更改和 Mocap 信息。
6. Assets/Neuron/Scripts/Mocap/NeuronAnimatorInstance.cs
继承自 NeuronInstance。提供自定义方法将 Axis 动捕数据直接应用于 Unity 的 Animator 组件中绑定骨骼的 transform 组件中，仅适用于人形骨骼。
7. Assets/Neuron/Scripts/Mocap/NeuronAnimatorPhysicalReference.cs
数据类，用于初始化和清理用于基于以下内容进行运动的参考骨骼 Unity 的刚体组件。如果启用了物理切换，则由 NeuronAnimatorInstance 使用。
8. Assets/Neuron/Scripts/Mocap/NeuronTransformsInstance.cs
继承自 NeuronInstance。提供自定义方法将 Axis 动捕数据直接应用于 transform 组件。适用于非人型骨架的模型或骨骼数量多于 Unity 默认骨骼数量的模型。
9. Assets/Neuron/Scripts/Mocap/NeuronTransformsPhysicalReference.cs
数据类，用于初始化和清理用于基于以下内容进行运动的参考骨骼 Unity 的刚体组件。如果启用了物理切换，则由 NeuronTransformsInstance 使用。
10. Assets/Neuron/Scripts/Mocap/NeuronInstancesManager.cs
对于每个 NeuronActor， NeuronInstancesManager 都会保留一个 NeuronAnimatorInstance，会在 NeuronDebugViewer 中使用。
11. Assets/Neuron/Scripts/Utilities/BoneLine.cs
使用线渲染器绘制骨骼线的实用程序类。
12. Assets/Neuron/Scripts/Utilities/BoneLines.cs
Neuron 编辑器的实用程序类，用于添加或删除 BoneLines。
13. Assets/Neuron/Scripts/Utilities/BoneRigidbody.cs
Neuron 编辑器的实用程序类，用于添加或删除刚体。
14. Assets/Neuron/Scripts/Utilities/FPSCounter.cs

用于计算 FPS（每秒帧数）的实用程序类。

3.3 NeuronTransformInstance.cs和NeuronAnimatorInstance.cs 的区别

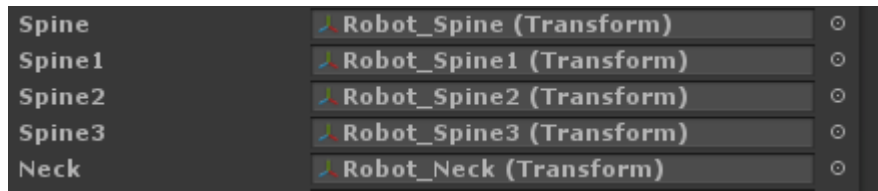
使用 NeuronAnimatorInstance.cs 脚本要求模型为人型骨架，如果导入的模型无法使用 Unity 提供的“人体骨骼（Humanoid）”系统，就需要使用 NeuronTransformInstance.cs，该脚本是通过 NeuronBones 名称（请参阅附录 B：骨架骨骼）进行每根骨骼以及动画的重定向的。



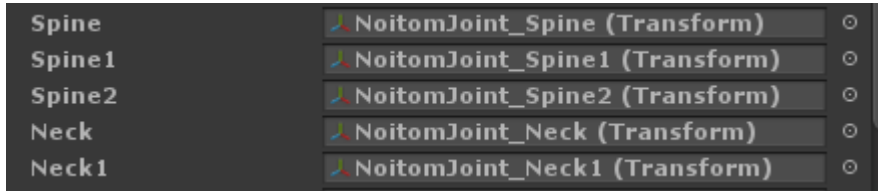
3.4 NeuronAnimatorInstance脚本中公共变量说明

1. Address: 是运行 Axis 软件所在的计算机的 IP 地址。如果是同一台计算机，可以使用 127.0.0.1。如果不在同一台计算机上请相应地更改地址。
2. Port: Axis 软件中的 BVH 数据流输出的端口号。
3. Socket Type: 网络协议选择，应该与 Axis 软件中设置相同。
4. Actor ID: 是您要使用的演员的 ID 号。如果在 Axis 软件中连接了多个角色，则此 ID 号将增加。默认值为 0 表示接收角色列表中的第一个角色数据。
5. Connect To Axis: 可以使用此开关来启动/停止数据流的接收。
6. Use New Rig: 不选中时程序会应用 AxisNeuron/AxisNeuronPro 骨骼，Spine 和 Neck 的映射方

式如下:



如果选中则应用 PerceptionNeuronStudio 骨骼, Spine 和 Neck 映射方式如下:



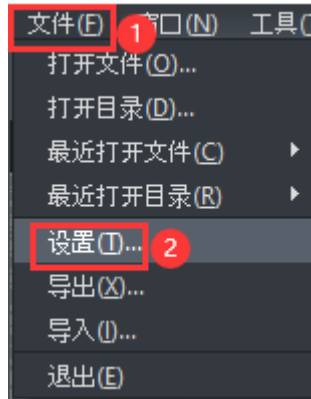
注: 两种骨骼定义参考附录 B:骨骼附录

7. Bound Animator: 可以设置将动画数据应用到哪个角色上, 如果运行前设置为空, 动画数据会应用到 Animator component 中的角色, 如果指定了角色, 那就会应用到指定角色上。
8. Motion Update Method: 是否使用 Unity 提供的刚体功能来移动和旋转每个骨骼。默认方法是将接收到的 float 值直接应用于每个骨骼的变换分量。

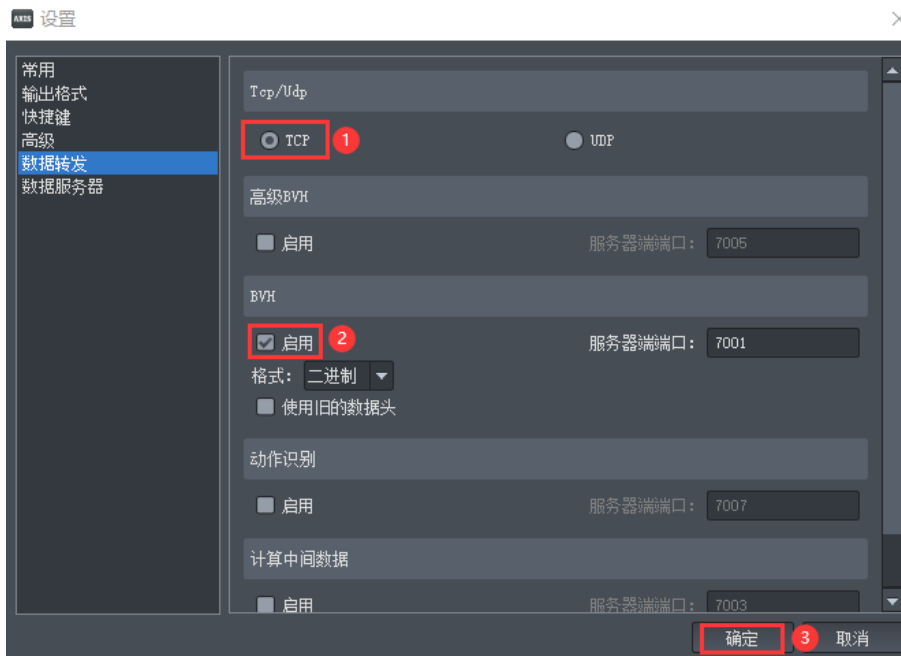
4. 如何在 Axis 软件中设置数据流转发

4.1 Axis Neuron/Axis Neuron Pro 软件设置

1. 启动 Axis Neuron/Axis Neuron Pro 软件
2. 点击文件 > 设置 > 数据转发



3. 选择 TCP/UDP, 该示例中以 TCP 为例, 打开 BVH 数据流开关, 其他设置保持默认然后点击【确定】按钮



4. 播放一个文件或者连接设备

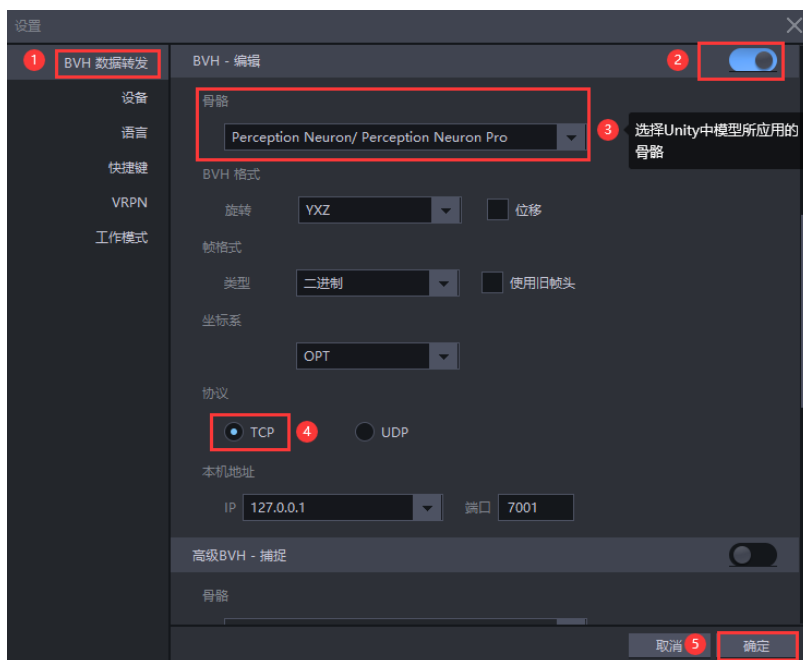
4.2 Axis Studio 软件设置

1. 启动 Axis Studio 软件

2. 点击主菜单 >设置 >BVH 数据转发

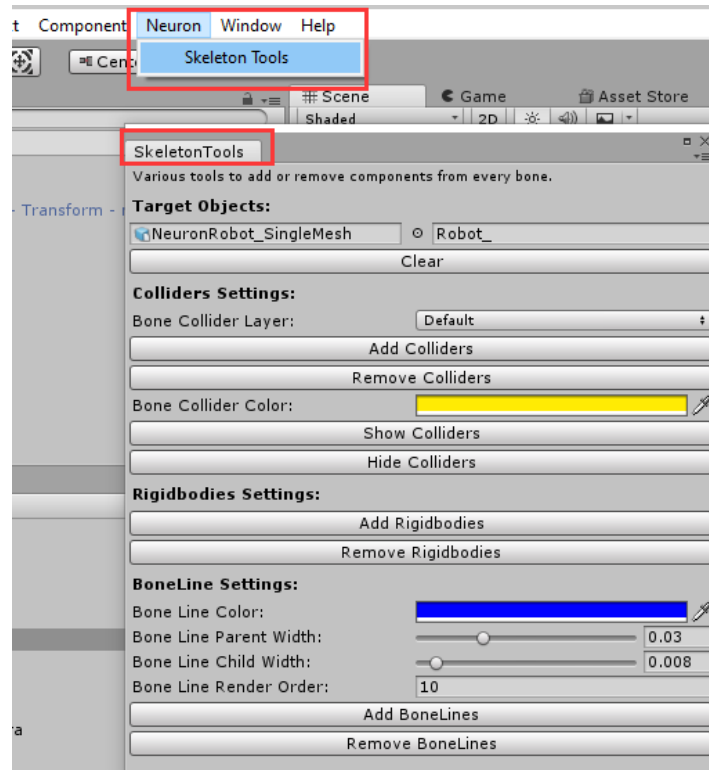


3. 选择数据源并打开该数据源开关，Axis studio 支持转发实时数据和离线数据，如
4. 转发离线数据需要选择 BVH-编辑，如转发实时数据需要选择 BVH-捕捉，该示例中以 BVH-编辑为例，选择 TCP/UDP 协议，该示例中以 TCP 为例，其他设置保持默认，设置完成后点击【确定】按钮



5. Neuron Skeleton Tools

NeuronSkeletonTools 是一个可以帮你配置和更改模型的工具，在菜单 Neuron-> Skeleton Tools 中。

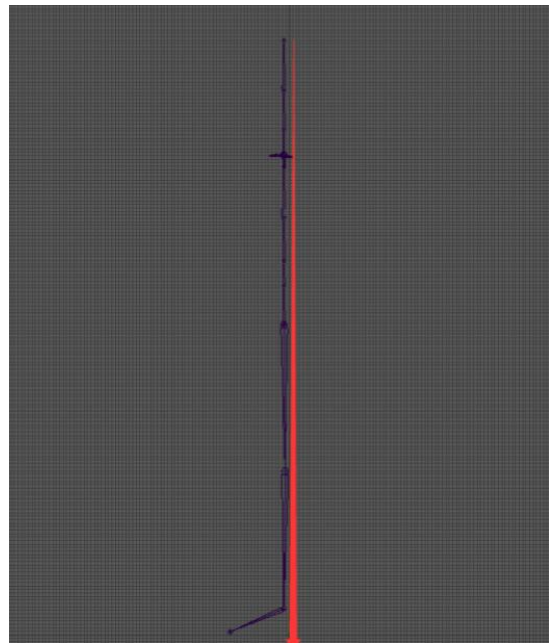
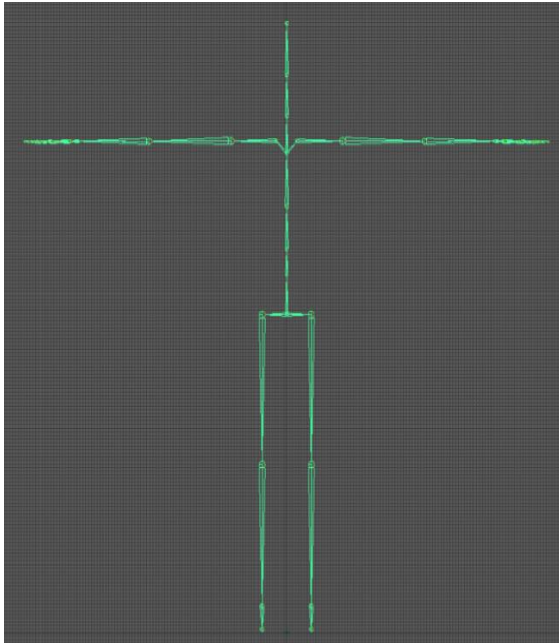


使用说明:

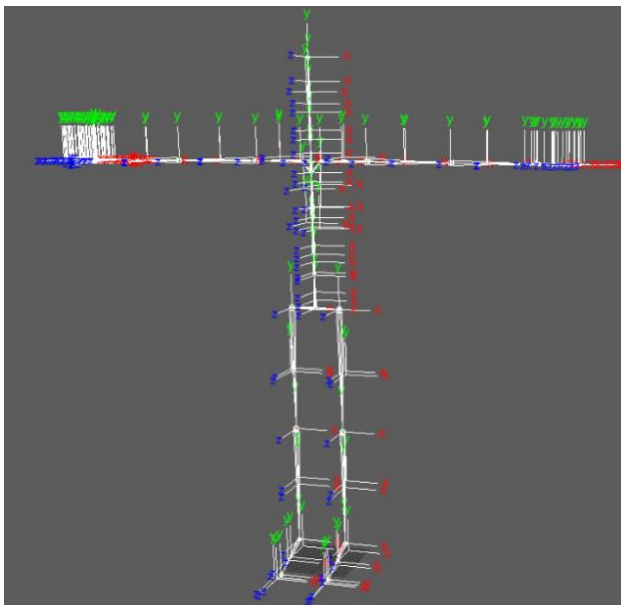
1. 首先要确保您选择的游戏对象附加了 NeuronTransformInstance 脚本
2. 支持添加/删除 Rigidbodies
3. 支持添加/删除 BoneLine。(BoneLine 是调试的绝佳工具，使用物理更新时，在运行时将其添加到实例化参考骨架中。Bone Line Render Order 定义了如果您有多条 Bone Line 时，哪条在顶部渲染。

6. 如何配置/选择动捕模型

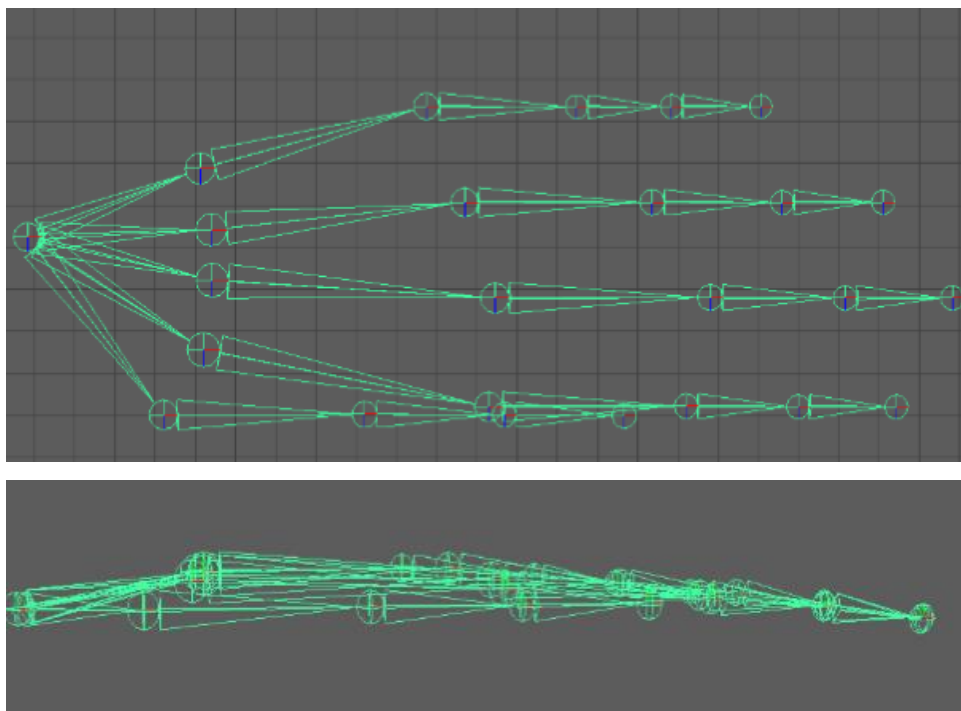
1. 使用人形骨骼的模型
2. 模型根节点旋转值必须为 0 值
3. 模型初始姿态最好为 T pose，具体要求如下：
 - 1) 从前视图看骨骼是 T Pose，从侧视图看骨骼是一条直线，如下图所示：



- 2) 所有骨骼的轴向朝向一致，如下图所示：



3) 手掌拇指并拢，其俯视图和正视图如下图所示：



4. Noitom 动捕系统支持两套骨骼，PN/PN Pro 骨骼和 PN Studio 骨骼，参考附录 B: 骨骼附录
5. 为了避免模型应用动捕数据后出现形变问题建议在 Axis 中使用模型的骨骼尺寸计算姿态，或者在 Axis 软件中设置数据流输出不使用带位移的数据。在 Axis 软件数据流设置页面取消勾选位移即可，取消发送位移数据并不是所有骨骼的数据不带位移数据，只是除 Hip 骨骼外其他骨骼不带位移数据。

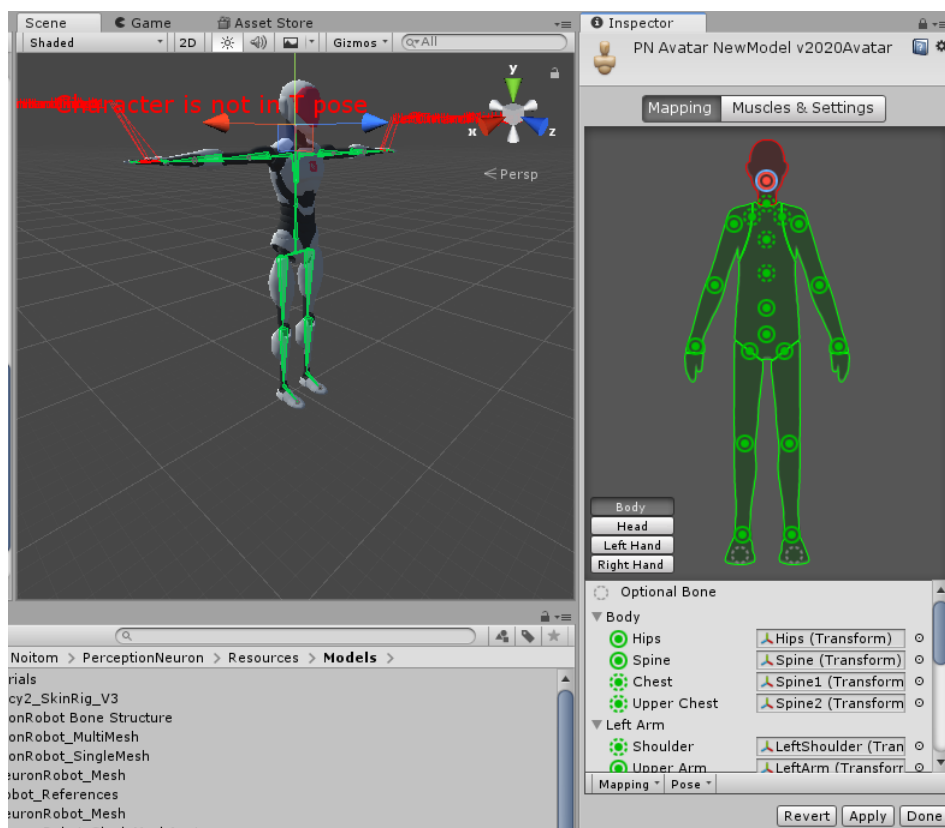
注释：如果 Axis 动画数据应用的所使用的模型上后效果不好，请先检测模型是否满足以上提到的几点要求，如果不满足请修改你的模型，最坏的情况就是重新建模，把模型装配到我们的标准骨骼上，标准骨骼的模型所在位置：[Neuron/Resources/Models/Template Bone Structure.fbx](#)

7. 如何使用 Axis 数据流实时驱动 Unity 中模型

1. 导入 PerceptionNeuron.unitypackage 资源包，如何导入可以参考 Unity 动捕资源包导入
2. 需要使用一个 Humanoid 人形骨骼的模型，模型要求参考如何配置/选择动捕模型，将模型导入到 Unity，该示例中将使用 AxisStudio 软件示例模型
3. Asset 中选中导入的模型，在 Unity 的“项目”选项卡中，单击您的模型，然后在检查器窗口中单击“Rig”选项卡，将您的动画类型设置为“人型 (Humanoid)”，然后单击【应用】

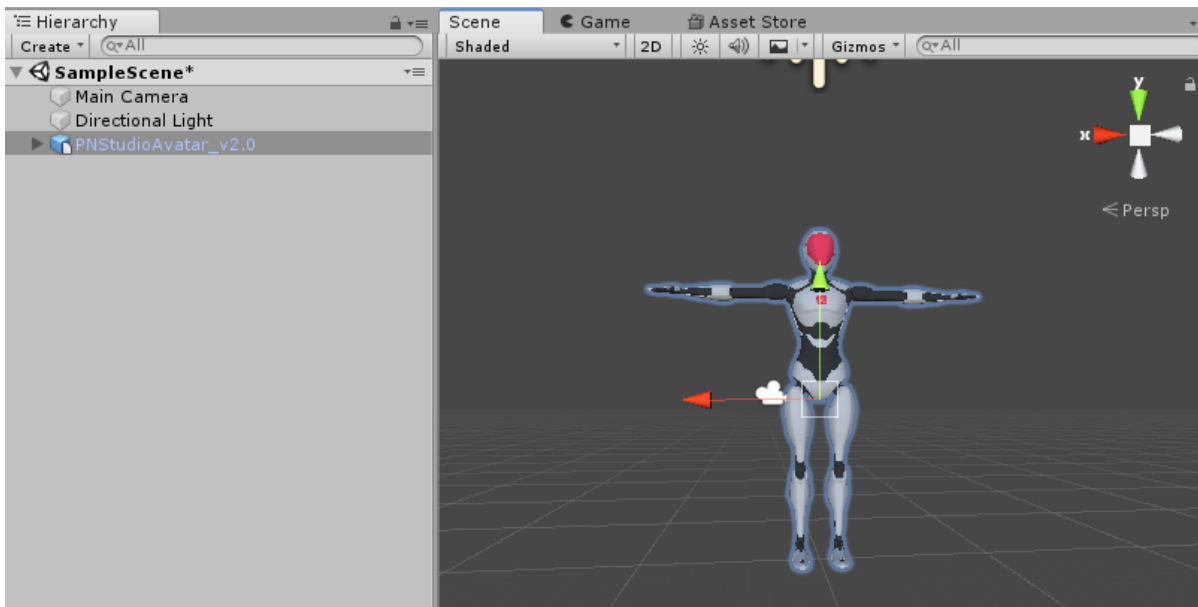


4. 单击【配置】以确保 Avatar 有效且正确设置。角色的骨骼结构匹配 Mecanim 预定义的骨骼结构且模型摆成 T 字姿势 (T-pose), 这一点非常重要。单击【配置】后看到一个带着骨骼映射的新 Avatar 配置监视器，如图所示：

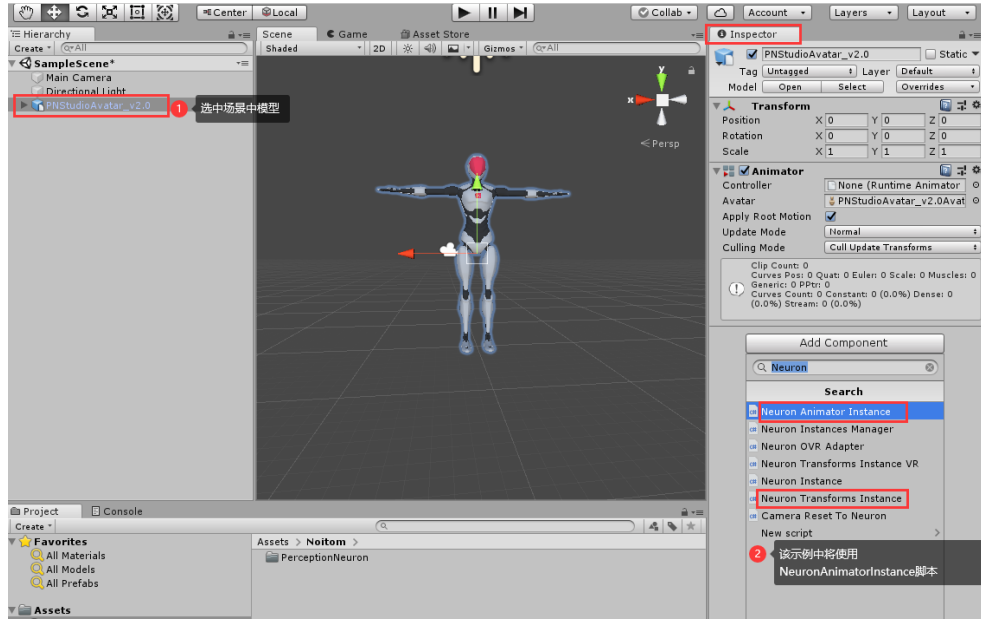


注释：如果自动映射（映射 (Mapping)->自动映射 (Automap)）完全或者部分失败，则可以通过从场景 (Scene) 或层级视图 (Hierarchy) 中拖出骨骼来为其赋值。如果 Mecanim 认为一块骨骼合适，该骨骼会在 Avatar 检视器 (Avatar Inspector) 中显示为绿色，否则显示为红色。最后，如果骨骼赋值正确，但角色的姿势错误，则会看到消息“角色未摆成 T 字姿势 (Character not in T-Pose)”。可使用强制 T 字姿势 (Enforce T-Pose) 进行修改，或者将剩余的骨骼旋转成 T 字姿势。

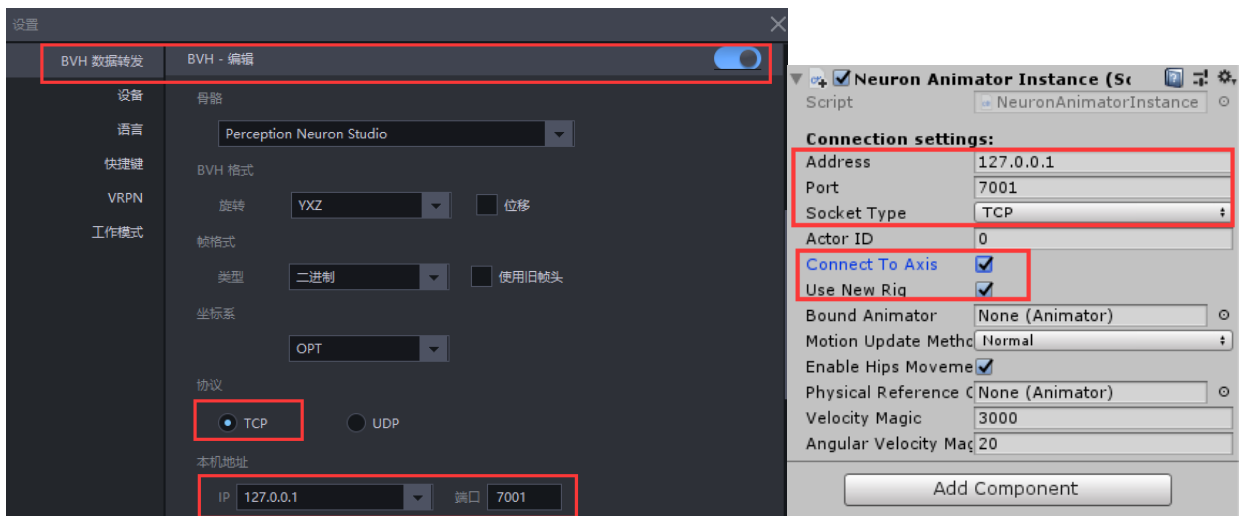
5. 可以点击 Mapping > Save 保存骨架中骨骼与 Avatar 的映射关系（扩展名为 *.ht），这些文件可以由使用此映射的任何角色重复使用。这十分有用，例如当动画对所有骨架使用一致的布局 and 命名约定，但 Mecanim 不知道如何解释它时。随后可以对每个模型加载 (Load) .ht 文件，以便手动重新映射只需进行一次。
6. 将导入的模型添加“层级结构 (Hierarchy)”中，如图所示：



7. Hierarchy 中选中选中模型并到 Inspector 面板中将 NeuronAnimatorInstance.cs 脚本附加到模型上，如果要使用完整的骨骼结构，可以使用 NeuronTransformInstance.cs

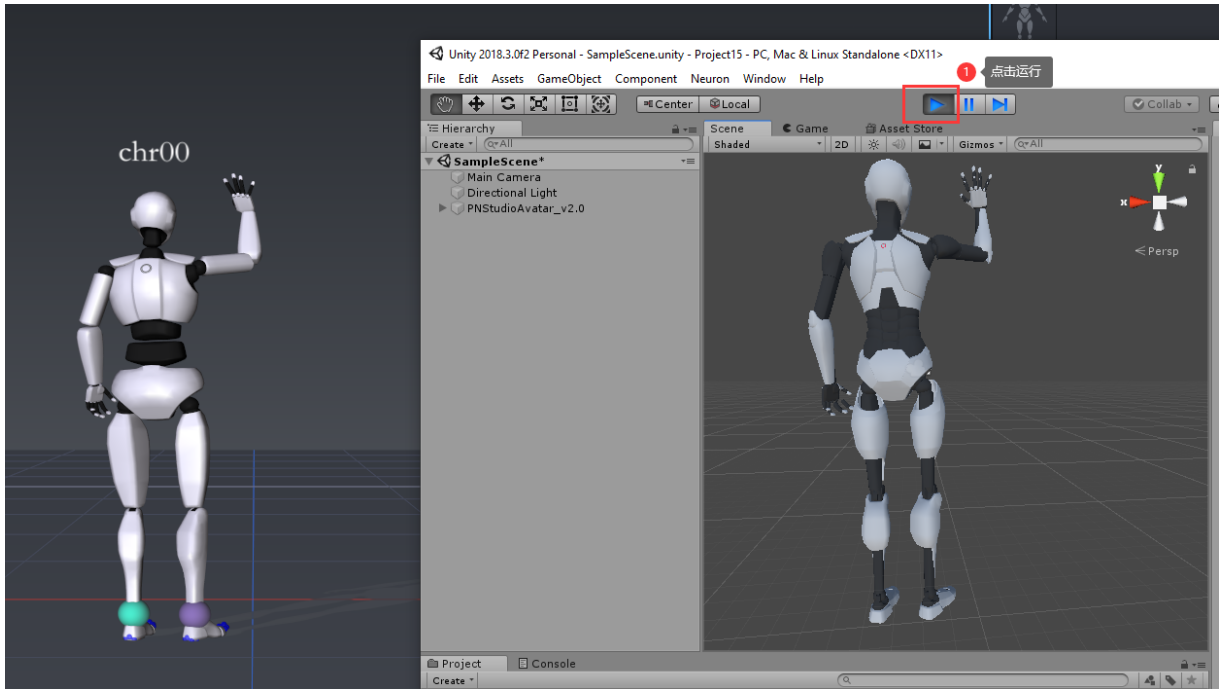


8. 根据 Axis 软件中设置的数据流参数完成 Connection Settings 设置，Axis 软件数据流转发设置参考如何在 Axis 软件中设置数据流转发, 本示例数据源将使用 AxisStudio 软件回放数据, 建议 AxisStudio 软件中循环播放已有动捕文件，保证数据流不中断，数据流相关设置如图所示：

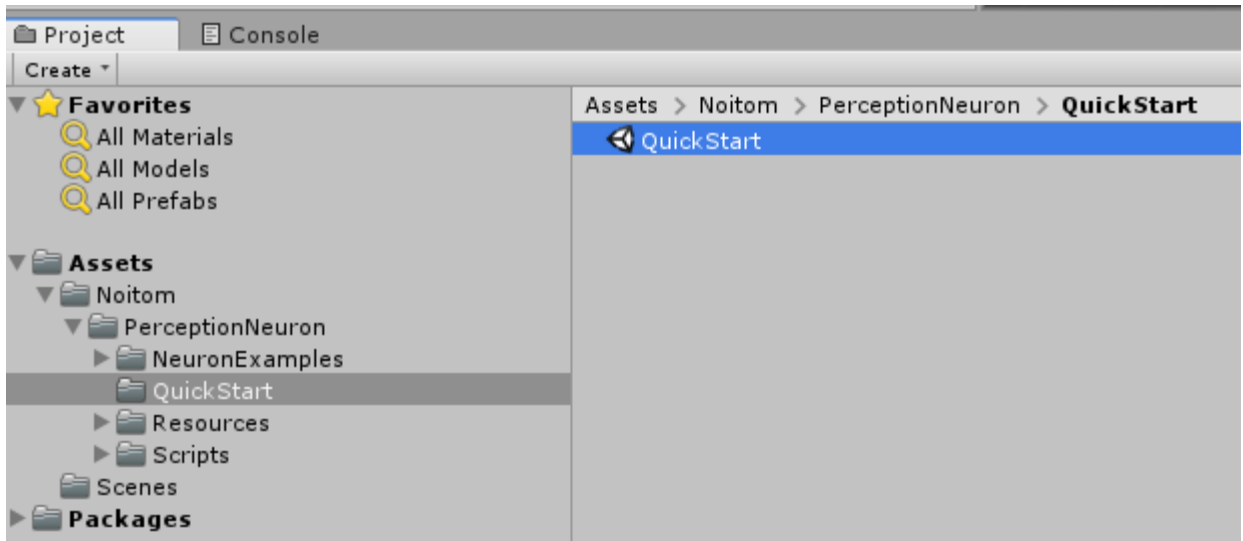


脚本中变量说明可参考：3.4 NeuronAnimatorInstance 脚本中公共变量说明

9. Unity 中点击【运行】，可查看当前模型被 Axis 数据实时驱动动的动画效果，如图所示：



注释：资源包中的 QuickStart 示例，包含了不同模型如何通过挂载提供的脚本来实现数据流驱动。



8. 头显集成

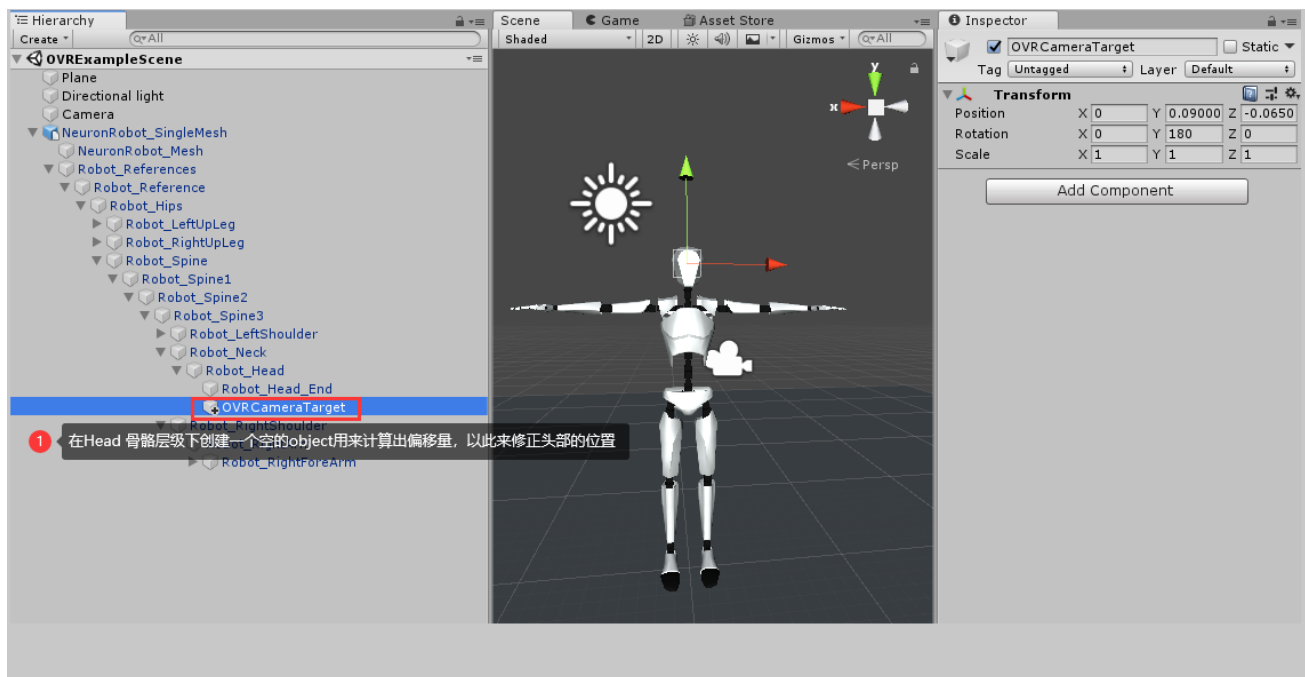
8.1 头显与Axis同步集成注意事项

1. 请使用头显自带的旋转（Rotation）值，不要使用头显的位置（Position）值
2. 不要在层级视图（Hierarchy）中选择头显或者游戏对象（GameObject）与骨骼模型应用父子化

8.2 示例脚本通过下列方式工作

在 NeuronExamples 文件夹中有一个名为 OVRExample 的示例。它包含一个简单的脚本和一个场景，展示如何将动捕系统与任何虚拟现实头显相结合，示例中包含的脚本的工作原理如下：

1. 每帧数据都会将 VR Camera 装置的位置重置到头部对象所在的位置。
2. 通过在 Head 层级下面创建一个空的 object（Hierarchy >Create Empty）来定义 offset 去校正 Head 位置。

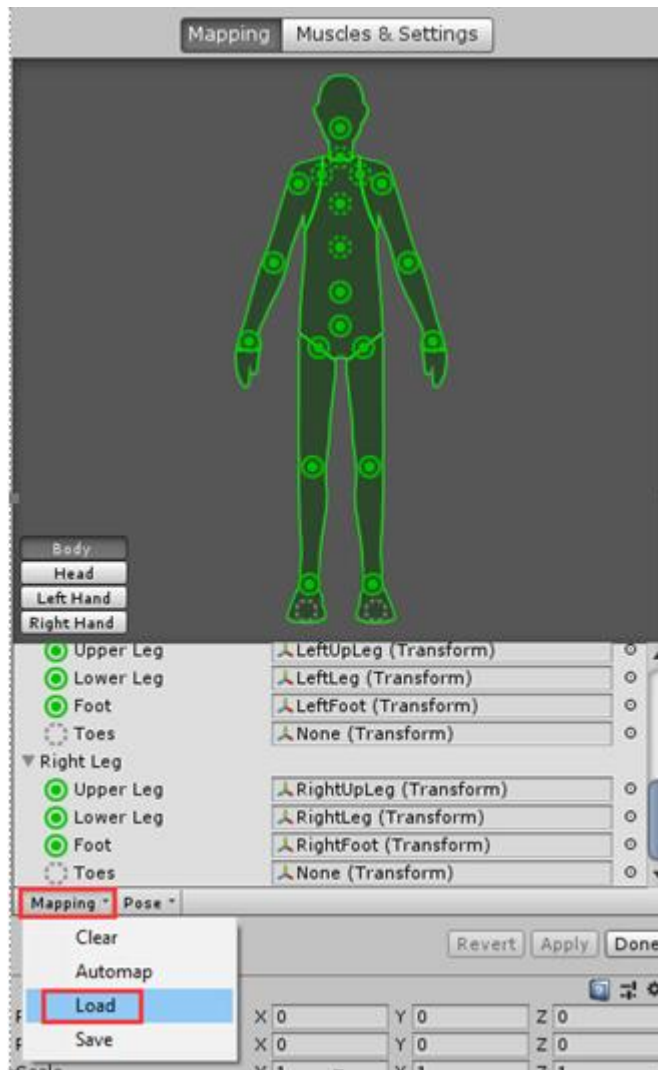


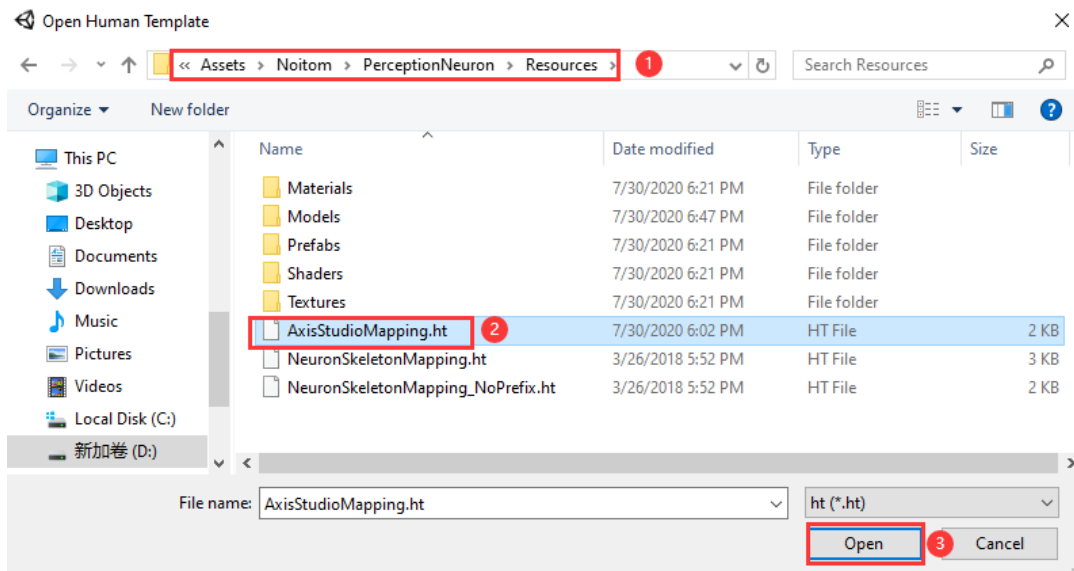
3. 戴上头显后我们要在保持 T-型姿态下，按键盘“R”键去重置头显以保证头显与虚拟角色模型进行初始化对齐。

9. 如何使用 AxisStudio 导出的 FBX 文件驱动模型

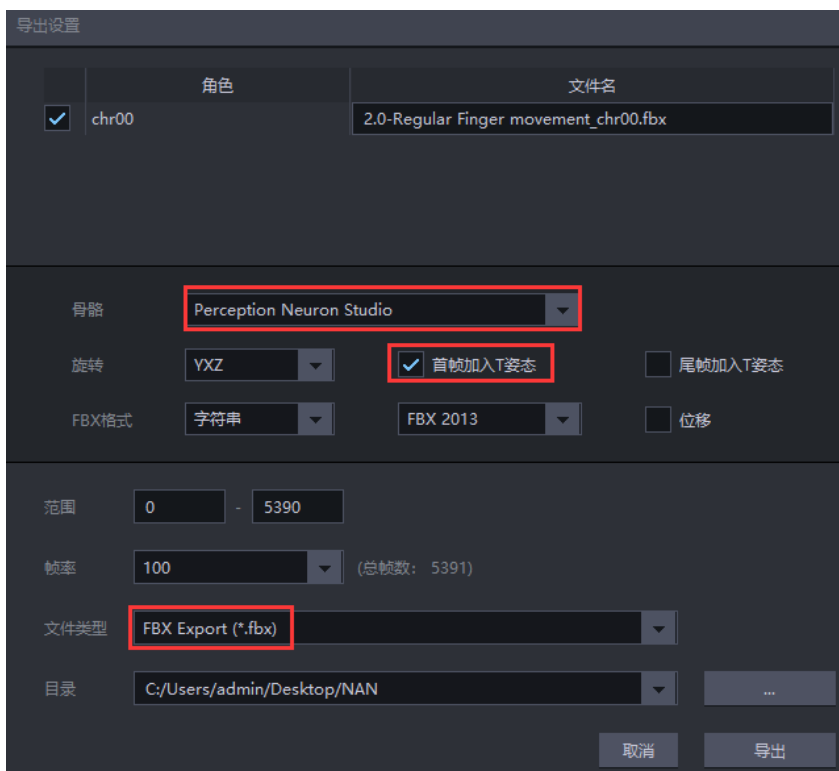
9.1 Humanoid 方式

1. 导入模型，在 Inspector 面板中将 Animation Type 设置为“Humanoid”，其他保持默认设置，设置完成后点击【Apply】应用
2. 点击【Configure】设置 Avatar 骨骼映射，可以直接加载我们预置好的骨骼映射文件，Mapping 面板点击 Mapping >Load AxisStudioMapping.ht 文件，如图所示：

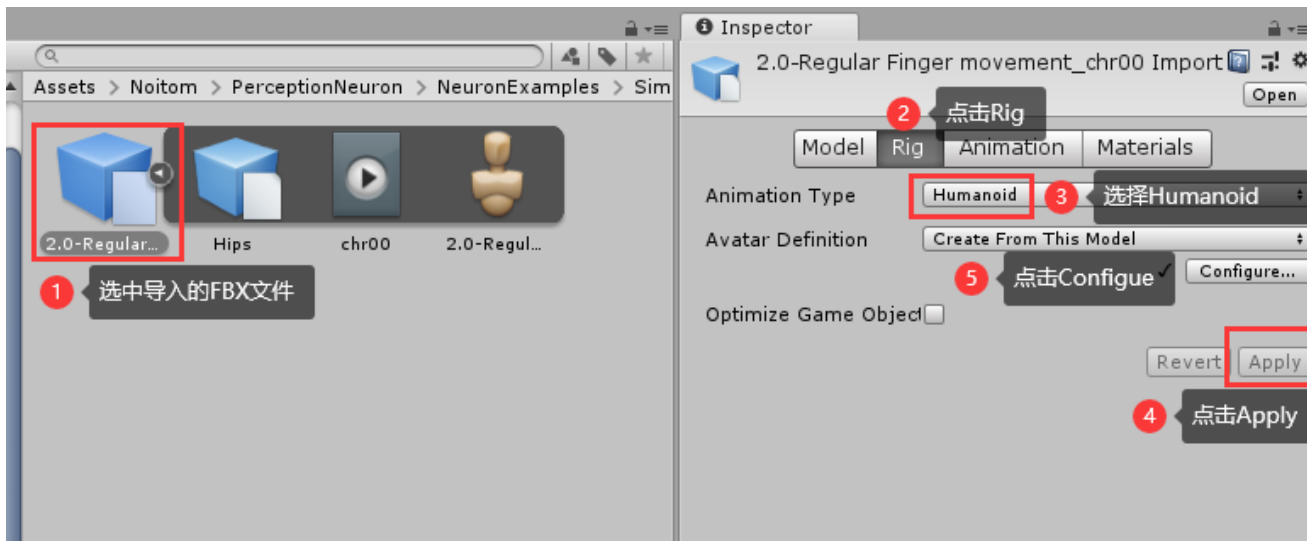




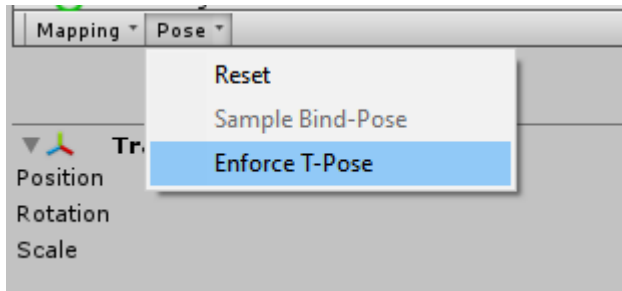
3. 然后点击【Apply】应用该映射
4. Hierachy 中依次选中每根骨骼，将其 Transform 中的 Rotation 中都清零，
5. 打开 AxisStudio 软件导出 FBX 文件，导出设置注意如下设置，由于该示例中使用的模型为 AxisStudio 的标准模型，所以导出设置时骨骼选择的是 PerceptionNeuronStudio，如果要使用的模型骨骼层级与 AxisNeuron 的一致，那么请选择 PerceptionNeuron 骨骼



6. 将 FBX 动画文件导入到当前打开的 Unity 工程中
7. 选中导入的文件，Inspector > Rig> Animation Type 选择 Humanoid 后先点击【Apply】再点击【Configure】，如图所示：



8. 打开 Avatar configuration 场景后，首先需要检测 character 是否为 Tpose，如果不是可以通过 Pose > Enforce T-pose 将其设置为 T-pose，如图所示：

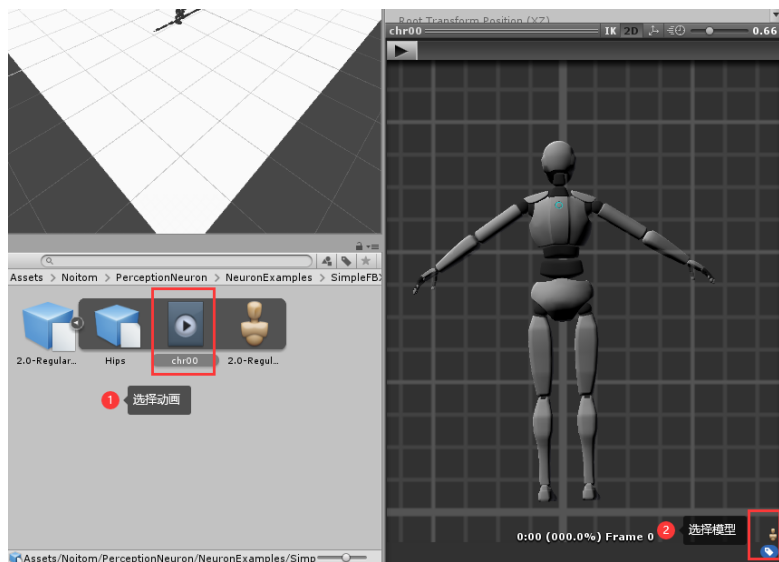


9. 检测骨骼映射关系是否正确，可以在 Mapping 面板点击 Mapping > Load ，加载 Resource 中预置的 AxisStudioMapping.ht 文件，该文件为预置好的与 AxisStudio 数据重定向的骨骼映射文件，操作同步骤 2，也可以手动修改映射关系。

注：如果使用的是 AxisNeuron 可以加载 NeuronSkeletonMapping.ht 文件，骨骼映射请参考[附录 A: Noitom 骨骼映射](#)

10. Hierachy 中依次选中每根骨骼，将其 Transform 中的 Rotation 中都清零，尤其是大拇指，目前 AxisStudio 导出的 fbx 导入的 Unity 中创建的角色的大拇指初始姿态与标准模型不一致，所以需要将大拇指 Rotation 设置为零，设置为零后与标准模型的初始姿态就一致了。
11. 以上设置完成后点击【Apply】应用

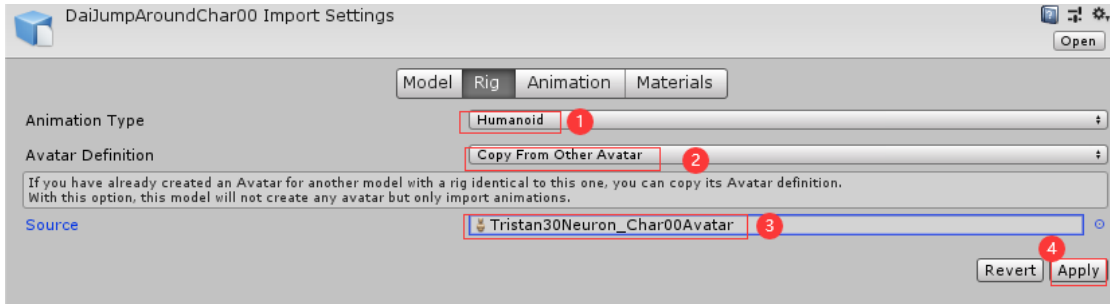
12. 点击【Done】返回上一场景
13. 选中动画，打开动画预览界面，点击右下脚可以选择模型，模型选择刚刚设置 Humanoid 的模型，如图所示：



14. 后点击【Play】就可以预览该动画了



注：如果动画数据是从 AxisNeuron 软件导出的，将 fbx 动画文件导入到 Unity 后，Rig > Avatar Definition 选择 “Copy From Other Avatar”，Source 选择 “Tristan30Neuron_Char00Avatar”，如图所示：



9.2 Generic 方式

Unity 中定义的 Generic 类型要求模型和动画文件中的骨骼名称和层级结构完全一致，如果你的模型骨骼层级结构和命名与 Axis 导出 FBX 文件的完全一致，那么你可以选择 Generic 方式，将 Rig > Animation type 设置为 Generic，Avatar Definition 设置为 Create From This Model 即可。

注释：Axis 导出的 FBX 文件中定义的骨骼层级和命名一致的参考模型位置：

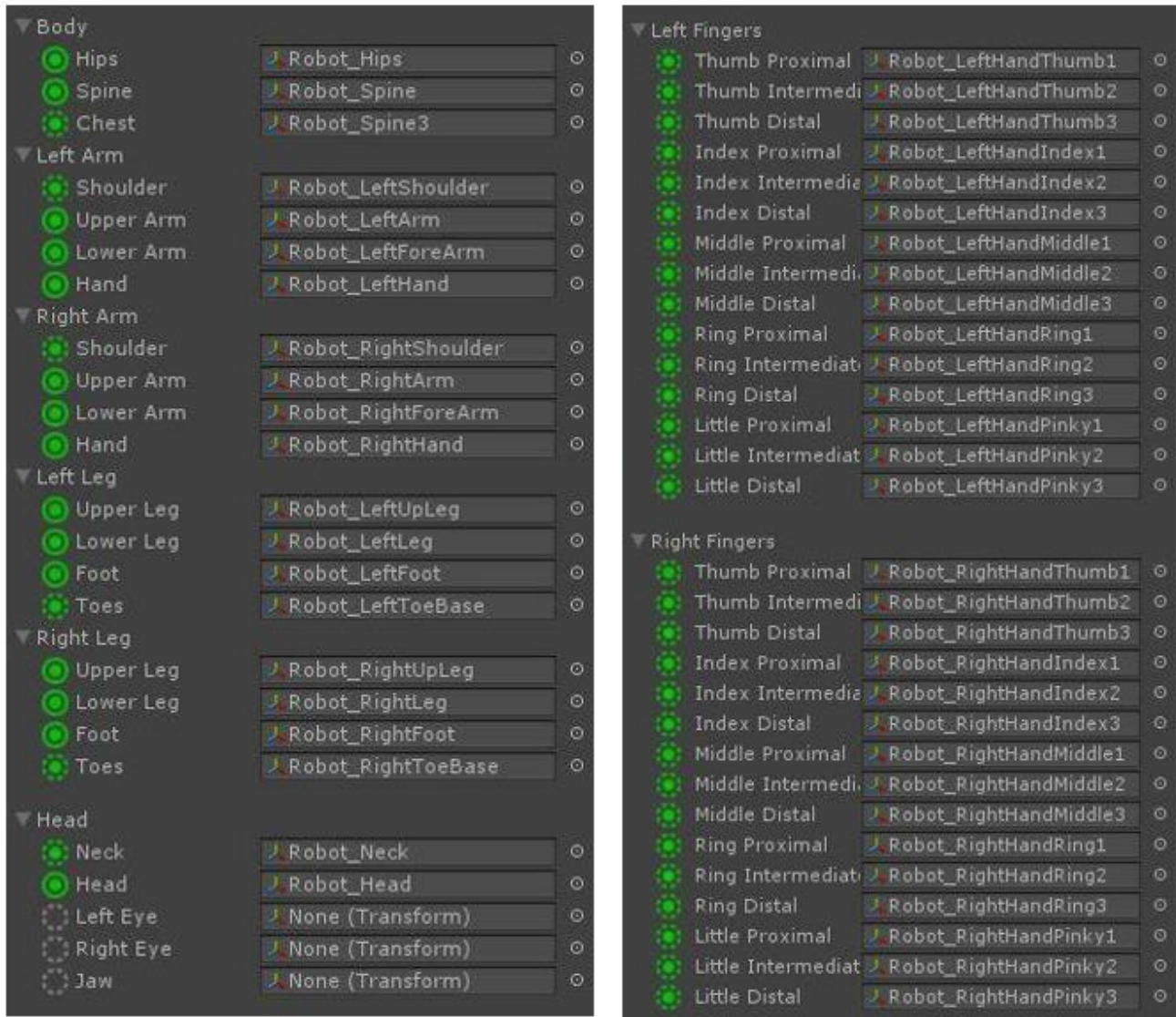
Neuron/Resources/Models/PNStudio_Avatar_SingleMesh.fbx 。

10. 附录 A: Noitom 骨骼映射

10.1 PerceptionNeuronStudio 骨骼映射

Head		Left Fingers	
● Neck	▲ Neck (Transform)	○ ● Thumb Proximal	▲ LeftHandThumb1 (Transform)
● Head	▲ Head (Transform)	○ ● Thumb Intermediate	▲ LeftHandThumb2 (Transform)
○ Left Eye	▲ None (Transform)	○ ● Thumb Distal	▲ LeftHandThumb3 (Transform)
○ Right Eye	▲ None (Transform)	○ ● Index Proximal	▲ LeftHandIndex1 (Transform)
○ Jaw	▲ None (Transform)	○ ● Index Intermediate	▲ LeftHandIndex2 (Transform)
▼ Body		○ ● Index Distal	▲ LeftHandIndex3 (Transform)
● Hips	▲ Hips (Transform)	○ ● Middle Proximal	▲ LeftHandMiddle1 (Transform)
▼ ● Spine	▲ Spine (Transform)	○ ● Middle Intermediate	▲ LeftHandMiddle2 (Transform)
● Chest	▲ Spine1 (Transform)	○ ● Middle Distal	▲ LeftHandMiddle3 (Transform)
● Upper Chest	▲ Spine2 (Transform)	○ ● Ring Proximal	▲ LeftHandRing1 (Transform)
▼ Left Arm		○ ● Ring Intermediate	▲ LeftHandRing2 (Transform)
● Shoulder	▲ LeftShoulder (Transform)	○ ● Ring Distal	▲ LeftHandRing3 (Transform)
▼ ● Upper Arm	▲ LeftArm (Transform)	○ ● Little Proximal	▲ LeftHandPinky1 (Transform)
● Lower Arm	▲ LeftForeArm (Transform)	○ ● Little Intermediate	▲ LeftHandPinky2 (Transform)
● Hand	▲ LeftHand (Transform)	○ ● Little Distal	▲ LeftHandPinky3 (Transform)
▼ Right Arm		Right Fingers	
● Shoulder	▲ RightShoulder (Transform)	○ ● Thumb Proximal	▲ RightHandThumb1 (Transform)
▼ ● Upper Arm	▲ RightArm (Transform)	○ ● Thumb Intermediate	▲ RightHandThumb2 (Transform)
● Lower Arm	▲ RightForeArm (Transform)	○ ● Thumb Distal	▲ RightHandThumb3 (Transform)
● Hand	▲ RightHand (Transform)	○ ● Index Proximal	▲ RightHandIndex1 (Transform)
○ Lower Arm	▲ RightForeArm (Transform)	○ ● Index Intermediate	▲ RightHandIndex2 (Transform)
○ Hand	▲ RightHand (Transform)	○ ● Index Distal	▲ RightHandIndex3 (Transform)
▼ Left Leg		○ ● Middle Proximal	▲ RightHandMiddle1 (Transform)
● Upper Leg	▲ LeftUpLeg (Transform)	○ ● Middle Intermediate	▲ RightHandMiddle2 (Transform)
● Lower Leg	▲ LeftLeg (Transform)	○ ● Middle Distal	▲ RightHandMiddle3 (Transform)
● Foot	▲ LeftFoot (Transform)	○ ● Ring Proximal	▲ RightHandRing1 (Transform)
● Toes	▲ LeftFoot_End (Transform)	○ ● Ring Intermediate	▲ RightHandRing2 (Transform)
▼ Right Leg		○ ● Ring Distal	▲ RightHandRing3 (Transform)
● Upper Leg	▲ RightUpLeg (Transform)	○ ● Little Proximal	▲ RightHandPinky1 (Transform)
● Lower Leg	▲ RightLeg (Transform)	○ ● Little Intermediate	▲ RightHandPinky2 (Transform)
● Foot	▲ RightFoot (Transform)	○ ● Little Distal	▲ RightHandPinky3 (Transform)
● Toes	▲ RightFoot_End (Transform)		

10.2 PerceptionNeuron骨骼映射



11. 附录 B: 骨骼附录

Axis Neuron:

- | | | | |
|-----|--------------------------------|-----|-------------------------------|
| 1. | Hips [↵] | 31. | RightHandRing2 [↵] |
| 2. | RightUpLeg [↵] | 32. | RightHandRing3 [↵] |
| 3. | RightLeg [↵] | 33. | RightInHandPinky [↵] |
| 4. | RightFoot [↵] | 34. | RightHandPinky1 [↵] |
| 5. | LeftUpLeg [↵] | 35. | RightHandPinky2 [↵] |
| 6. | LeftLeg [↵] | 36. | RightHandPinky3 [↵] |
| 7. | LeftFoot [↵] | 37. | LeftShoulder [↵] |
| 8. | Spine [↵] | 38. | LeftArm [↵] |
| 9. | Spine1 [↵] | 39. | LeftForeArm [↵] |
| 10. | Spine2 [↵] | 40. | LeftHand [↵] |
| 11. | Spine3 [↵] | 41. | LeftHandThumb1 [↵] |
| 12. | Neck [↵] | 42. | LeftHandThumb2 [↵] |
| 13. | Head [↵] | 43. | LeftHandThumb3 [↵] |
| 14. | RightShoulder [↵] | 44. | LeftInHandIndex [↵] |
| 15. | RightArm [↵] | 45. | LeftHandIndex1 [↵] |
| 16. | RightForeArm [↵] | 46. | LeftHandIndex2 [↵] |
| 17. | RightHand [↵] | 47. | LeftHandIndex3 [↵] |
| 18. | RightHandThumb1 [↵] | 48. | LeftInHandMiddle [↵] |
| 19. | RightHandThumb2 [↵] | 49. | LeftHandMiddle1 [↵] |
| 20. | RightHandThumb3 [↵] | 50. | LeftHandMiddle2 [↵] |
| 21. | RightInHandIndex [↵] | 51. | LeftHandMiddle3 [↵] |
| 22. | RightHandIndex1 [↵] | 52. | LeftInHandRing [↵] |
| 23. | RightHandIndex2 [↵] | 53. | LeftHandRing1 [↵] |
| 24. | RightHandIndex3 [↵] | 54. | LeftHandRing2 [↵] |
| 25. | RightInHandMiddle [↵] | 55. | LeftHandRing3 [↵] |
| 26. | RightHandMiddle1 [↵] | 56. | LeftInHandPinky [↵] |
| 27. | RightHandMiddle2 [↵] | 57. | LeftHandPinky1 [↵] |
| 28. | RightHandMiddle3 [↵] | 58. | LeftHandPinky2 [↵] |
| 29. | RightInHandRing [↵] | 59. | LeftHandPinky3 [↵] |
| 30. | RightHandRing1 [↵] | | |

Axis Studio:

- | | | | |
|-----|--------------------|-----|-------------------|
| 1. | Hips↵ | 31. | RightHandRing2↵ |
| 2. | RightUpLeg↵ | 32. | RightHandRing3↵ |
| 3. | RightLeg↵ | 33. | RightInHandPinky↵ |
| 4. | RightFoot↵ | 34. | RightHandPinky1↵ |
| 5. | LeftUpLeg↵ | 35. | RightHandPinky2↵ |
| 6. | LeftLeg↵ | 36. | RightHandPinky3↵ |
| 7. | LeftFoot↵ | 37. | LeftShoulder↵ |
| 8. | Spine↵ | 38. | LeftArm↵ |
| 9. | Spine1↵ | 39. | LeftForeArm↵ |
| 10. | Spine2↵ | 40. | LeftHand↵ |
| 11. | Neck↵ | 41. | LeftHandThumb1↵ |
| 12. | Neck1↵ | 42. | LeftHandThumb2↵ |
| 13. | Head↵ | 43. | LeftHandThumb3↵ |
| 14. | RightShoulder↵ | 44. | LeftInHandIndex↵ |
| 15. | RightArm↵ | 45. | LeftHandIndex1↵ |
| 16. | RightForeArm↵ | 46. | LeftHandIndex2↵ |
| 17. | RightHand↵ | 47. | LeftHandIndex3↵ |
| 18. | RightHandThumb1↵ | 48. | LeftInHandMiddle↵ |
| 19. | RightHandThumb2↵ | 49. | LeftHandMiddle1↵ |
| 20. | RightHandThumb3↵ | 50. | LeftHandMiddle2↵ |
| 21. | RightInHandIndex↵ | 51. | LeftHandMiddle3↵ |
| 22. | RightHandIndex1↵ | 52. | LeftInHandRing↵ |
| 23. | RightHandIndex2↵ | 53. | LeftHandRing1↵ |
| 24. | RightHandIndex3↵ | 54. | LeftHandRing2↵ |
| 25. | RightInHandMiddle↵ | 55. | LeftHandRing3↵ |
| 26. | RightHandMiddle1↵ | 56. | LeftInHandPinky↵ |
| 27. | RightHandMiddle2↵ | 57. | LeftHandPinky1↵ |
| 28. | RightHandMiddle3↵ | 58. | LeftHandPinky2↵ |
| 29. | RightInHandRing↵ | 59. | LeftHandPinky3 |
| 30. | RightHandRing1↵ | | |

以上是完整的 59 根骨骼序列，如果在 Unity 中使用的模型是 Humanoid 骨骼，我们可以忽略掉骨骼名中包含 InHand 的骨骼，例如：序列 21 的 RightInHandIndex，也可以忽略掉两节 Spine 骨骼，去掉这 11 根个就意味着我们的只需要处理处理剩下的 49 根骨骼的映射即可。

12. 附录 C：二进制数据序列

Axis 接收的二进制数据的整个序列的完整图形，它是一维浮点数组，具有不同的顺序和长度，具体取决于您是否使用位移数据。

	NO DISPLACEMENT						WITH DISPLACEMENT					
	Position			Rotation			Position			Rotation		
	X	Y	Z	Y	X	Z	X	Y	Z	Y	X	Z
Bone												
Hips	0	1	2	3	4	5	0	1	2	3	4	5
RightUpLeg				6	7	8	6	7	8	9	10	11
RightLeg				9	10	11	12	13	14	15	16	17
RightFoot				12	13	14	18	19	20	21	22	23
LeftUpLeg				15	16	17	24	25	26	27	28	29
LeftLeg				18	19	20	30	31	32	33	34	35
LeftFoot				21	22	23	36	37	38	39	40	41
Spine				24	25	26	42	43	44	45	46	47
Spine1				27	28	29	48	49	50	51	52	53
Spine2				30	31	32	54	55	56	57	58	59
Spine3				33	34	35	60	61	62	63	64	65
Neck				36	37	38	66	67	68	69	70	71
Head				39	40	41	72	73	74	75	76	77
RightShoulder				42	43	44	78	79	80	81	82	83
RightArm				45	46	47	84	85	86	87	88	89
RightForeArm				48	49	50	90	91	92	93	94	95
RightHand				51	52	53	96	97	98	99	100	101
RightHandThumb1				54	55	56	102	103	104	105	106	107
RightHandThumb2				57	58	59	108	109	110	111	112	113
RightHandThumb3				60	61	62	114	115	116	117	118	119
RightInHandIndex				63	64	65	120	121	122	123	124	125
RightHandIndex1				66	67	68	126	127	128	129	130	131
RightHandIndex2				69	70	71	132	133	134	135	136	137
RightHandIndex3				72	73	74	138	139	140	141	142	143
RightInHandMiddle				75	76	77	144	145	146	147	148	149
RightHandMiddle1				78	79	80	150	151	152	153	154	155
RightHandMiddle2				81	82	83	156	157	158	159	160	161
RightHandMiddle3				84	85	86	162	163	164	165	166	167
RightInHandRing				87	88	89	168	169	170	171	172	173
RightHandRing1				90	91	92	174	175	176	177	178	179
RightHandRing2				93	94	95	180	181	182	183	184	185
RightHandRing3				96	97	98	186	187	188	189	190	191
RightInHandPinky				99	100	101	192	193	194	195	196	197
RightHandPinky1				102	103	104	198	199	200	201	202	203
RightHandPinky2				105	106	107	204	205	206	207	208	209
RightHandPinky3				108	109	110	210	211	212	213	214	215

AxisStudio

Bone	NO DISPLACEMENT						WITH DISPLACEMENT					
	Position			Rotation			Position			Rotation		
	X	Y	Z	X	Y	Z	X	Y	Z	X	Y	Z
Hips	0	1	2	3	4	5	0	1	2	3	4	5
RightUpLeg				6	7	8	6	7	8	9	10	11
RightLeg				9	10	11	12	13	14	15	16	17
RightFoot				12	13	14	18	19	20	21	22	23
LeftupLeg				15	16	17	24	25	26	27	28	29
LeftLeg				18	19	20	30	31	32	33	34	35
Left Foot				21	22	23	36	37	38	39	40	41
Spine				24	25	26	42	43	44	45	46	47
Spine1				27	28	29	48	49	50	51	52	53
Spine2				30	31	32	54	55	56	57	58	59
Neck				33	34	35	60	61	62	63	64	65
Neck1				36	37	38	66	67	68	69	70	71
Head				39	40	41	72	73	74	75	76	77
RightShoulder				42	43	44	78	79	80	81	82	83
RightArm				45	46	47	84	85	86	87	88	89
RightForeArm				48	49	50	90	91	92	93	94	95
RightHand				51	52	53	96	97	98	99	100	101
RightHandThumb1				54	55	56	102	103	104	105	106	107
RightHandThumb2				57	58	59	108	109	110	111	112	113
RightHandThumb3				60	61	62	114	115	116	117	118	119
RightinHandindex				63	64	65	120	121	122	123	124	125
RightHandIndex1				66	67	68	126	127	128	129	130	131
RightHandIndex2				69	70	71	132	133	134	135	136	137
RightHandIndex3				72	73	74	138	139	140	141	142	143
RightinHandMiddle				75	76	77	144	145	146	147	148	149
RightHandMiddle1				78	79	80	150	151	152	153	154	155
RightHandMiddle2				81	82	83	156	157	158	159	160	161
RightHandMiddle3				84	85	86	162	163	164	165	166	167
RightinHandRing				87	88	89	168	169	170	171	172	173
RightHandRing1				90	91	92	174	175	176	177	178	179
RightHandRing2				93	94	95	180	181	182	183	184	185
RightHandRing3				96	97	98	186	187	188	189	190	191
RightHandPinky				99	100	101	192	193	194	195	196	197
RightHandPinky1				102	103	104	198	199	200	201	202	203
RightHandPinky2				105	106	107	204	205	206	207	208	209
RightHandPinky3				108	109	110	210	211	212	213	214	215