

poselib 是一个用于加载、操作和重新定位骨架姿势和动作的Python库

该库分为三个主要模块：

- `poselib.core` 用于数据加载和张量操作
- `poselib.skeleton` 用于更高级的骨架操作
- `poselib.visualization` 用于显示骨架姿势

poselib.core 模块提供了一些 Torch JIT(即时编译) 函数，用于计算四元数、变换和旋转/变换矩阵。以下是该模块中的一些主要功能：

rotation3d: 这部分包含了一组 Torch JIT 函数，用于计算四元数、变换和旋转/变换矩阵。这些函数可能用于处理三维旋转和变换操作，通常在动作捕捉、计算机图形学等领域中会经常用到。这些函数可能提供高性能的计算，以便在实时环境中使用。

quat_*: 这部分处理四元数的创建和操作。四元数通常用于表示旋转操作，其中 $[x, y, z, w]$ 表示四元数的分量，其中 w 是实部。这些函数可能包括创建四元数、进行乘法操作、插值等操作。

transform_*: 这部分处理七维变换，其中 $[quat, pos]$ 表示变换的分量，包括旋转和平移。这些函数可能用于组合四元数旋转和位置平移，以生成完整的变换操作。

rot_matrix_*: 这部分处理三行三列的旋转矩阵。旋转矩阵用于描述物体在三维空间中的旋转变换。这些函数可能用于创建、组合和操作旋转矩阵。

euclidean_*: 这部分处理四行四列的欧几里德变换矩阵。这种矩阵包括旋转、平移和缩放等变换操作。这些函数可能用于创建、组合和操作欧几里德变换矩阵。

tensor_utils: 这部分提供了用于加载和保存 PyTorch 张量的函数。这些函数可能用于将数据从外部文件加载到 PyTorch 张量中，或者将 PyTorch 张量保存到文件中。

总的来说，`poselib.core` 模块似乎提供了用于处理姿势和变换的核心功能，包括四元数、变换矩阵和旋转矩阵等操作。这些功能在计算机图形学、动作捕捉、机器人技术等领域中非常有用，能够帮助处理三维空间中的旋转和变换操作，并且通过 Torch JIT 函数提供高性能的计算。

poselib.skeleton 模块提供了加载、操作骨架姿势以及将姿势重新定位到不同骨架的工具。以下是该模块中的一些主要功能：

skeleton3d: 这部分提供了加载和操作骨架姿势的实用工具，还包括将姿势重新定位到不同骨架的功能。这些工具可能用于处理骨架动作捕捉数据，以及将姿势从一个骨架适应到另一个骨架的需求。

SkeletonTree: 这是一个类，将骨架存储为树状结构。它描述了骨架的拓扑结构和关节。这对于描述骨架的层次结构以及骨架中各个关节之间的关系非常有用。

SkeletonState: 这个类描述了骨架的静态状态，并提供了全局和局部关节角度。这可能是描述骨架姿势的一种方式，可以包括关节的旋转信息。

SkeletonMotion: 这个类描述了骨架状态的时间序列，并提供了计算关节速度等工具。这可能用于处理骨架的动态变化，比如骨架的运动。

poselib.visualization 模块提供了用于在 matplotlib 中以交互方式可视化骨架的函数。以下是该模块中的一些主要功能：

common: 这部分提供了用于在 matplotlib 中交互式可视化骨架的函数。这些函数可能用于将骨架姿势可视化出来，以便更好地理解骨架的结构和动态变化。
在骨架状态可视化中，按键 q 可以退出窗口。在交互式的骨架动作可视化中，你可以使用以下按键命令：

w: 循环播放动画
x: 播放/暂停动画
z: 前一帧
c: 后一帧
n: 退出窗口

总的来说，poselib.skeleton 模块提供了处理骨架姿势和动作的工具，而 poselib.visualization 模块则提供了在 matplotlib 中可视化这些骨架姿势和动作的功能。