视觉项目：虚拟人象棋游戏，通过openpose实现对玩家身体姿态信息的捕捉，得到结点信息的json文件，实现虚拟人动作同步（Unity项目）

玩家身体姿态的识别通过OpenPose模型检测出场景中所有人的身体姿势信息，内容包括关键点的位置、骨骼连接和主要姿势，识别主要包括以下几个步骤：

预处理：OpenPose加载实时视频帧，并对其进行预处理，将图像转换为合适的尺寸和颜色空间，以便后续的姿势估计。

人体关键点检测：使用深度学习:基于卷积神经网络（CNN）来检测人体的关键点来预测每个关键点的位置，包括头部、肩膀、手臂、手部、腿部等。这些关键点将被用作定义人体的姿势。

骨骼连接：在检测到所有关键点后，连接这些关键点来构建人体的骨骼。

姿势估计：将所有的关键点和骨骼连接结合在一起，从而得到完整的人体姿势。

在完成识别后，系统可以得到关键点在图像的具体位置，位置将以json文件的形式存储。系统通过识别Json文件，实现对识别信息的解析，使用JsonUtility.FromJson（）将读取到的JSON文本转换为ModelTest对象。ModelTest是一个自定义类，用来表示JSON数据的结构。对json文件的读取核心代码如下：

string jsonTest = File.ReadAllText(a, Encoding.UTF8);

ModelTest obj = JsonUtility.FromJson<ModelTest>(jsonTest);

for (int i = 0; i < 18; i++){

data[i] = new Vector2(obj.people[0].pose\_keypoints\_2d[3 \* i], obj.people[0].pose\_keypoints\_2d[3 \* i + 1]);

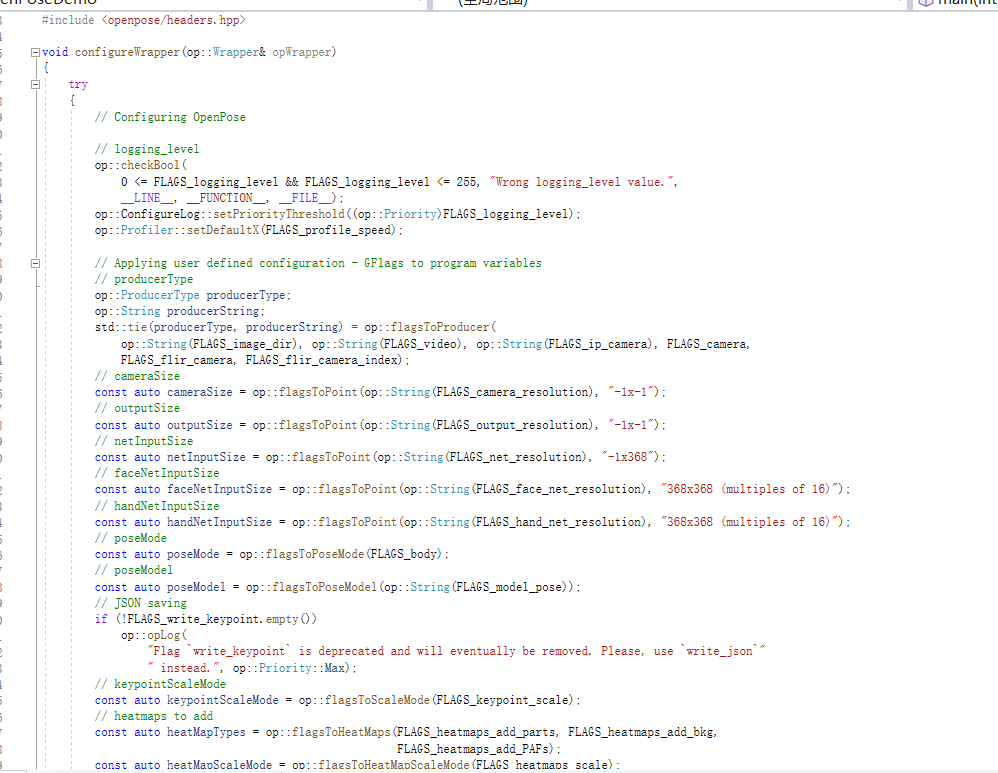
}

得到信息后，通过将信息解析，实现对对应关节节点的控制。

Codding:

通过一系列的配置步骤，将用户的命令行参数和配置选项应用于 OpenPose 的各个模块，确保 OpenPose 能够根据指定的参数正确地启动和运行:日志和性能配置,设置日志级别和性能分析工具。检查并设置日志级别，确保其在有效范围内。设置性能分析工具的默认行为。输入源配置,根据用户提供的参数确定输入源类型。设置摄像头分辨率和输出分辨率。网络输入和输出配置,网络输入尺寸、输出尺寸、面部和手部网络输入尺寸。设置姿态检测模式和模型。关键点和热图配置。设置关键点缩放模式和热图类型。设置热图缩放模式。多视图和检测器配置。检查是否启用多视图模式和面部和手部检测器。姿态检测配置：配置姿态检测模块，包括模式、网络输入尺寸、输出尺寸、关键点缩放模式、GPU设置、渲染模式等。

配置面部检测模块和手部检测配置，包括是否启用、检测器、网络输入尺寸、渲染模式等。

初始化和记录日志。配置 OpenPose 的各项参数。启动并运行 OpenPose。测量并记录总运行时间。处理异常情况。

