# 骨骼姿态识别算法工程化需求

## 总体描述

1、算法以http service形式提供API接口服务，调用方发起http post请求，使用json传递请求参数，服务端以json格式返回结果；

2、使用配置文件，与代码放置在同一文件夹下，用于设置服务IP、服务端口、视频/图片输入文件路径、临时文件夹路径；

3、三个算法名分别为SkeletonPose、One-leggedStanding、SwingFrequency；

4、三个算法服务名格式为算法名+版本号，如：skeletonpose/v1.0；

5、对于SkeletonPose，输出内容为每个人的鼻子、左眼、右眼、左耳、右耳、左肩、右肩、左肘、右肘、左手、右手、左胯、右胯、左膝、右膝、左脚、右脚共17个点的坐标（x,y）组成的数据结构的数组，视频/图片左上角坐标为（0，0），单位是像素；

6、对于One-leggedStanding、SwingFrequency，输入内容为单个受试者的受试视频，录制流程如下：1、受试者站立于屏幕中央（录像期间不要有其他人近身产生干扰）；2、开始录像；3、受试者开始单脚站立；4、受试者双脚着地，测试结束；5、结束录像；

## 骨骼姿态识别算法

输入

地址：http://IP:Port/ai/alphapose/skeletonpose/v1.0

Metohd：Post

参数：{

“subjectNumber”:”single”,//single或multiple，当参数为single时，仅识别画面中间区域，对应单人场景；当参数为multiple时，可以全屏识别出多人的骨骼姿态

“inputType”:”video”，//video或image，若输入为图片，需要按图片行为顺序命名，例如，“frame-0001.jpg”，“frame-0002.jpg”……，“frame-0001.jpg”图片的动作发生在“frame-0002.jpg”之前

“framePerSecond”:“-1“,//每秒抽取多少帧进行计算，-1为使用视频的原始帧率

}

输出格式：json

{

[

{“frame”:”frame\_xxx”,//本帧图片名

“pos1”:” 148.08267211914062, 170.41311645507812”,// posX代表位置如附图，posX的值表示为以屏幕左上角为坐标原点（0，0）的像素点的横坐标和纵坐标的值

”pos2”:…},

{“frame”:”frame\_xxx”,“pos1”:…}, //对于一个frame中识别出多人的情况，相同frame值会出现多条记录

…

]

}



## 单脚站立时长

输入

地址：http://IP:Port/ai/alphapose/onelegedstand/v1.0

Metohd：Post

参数：{

“inputType”:”video”，//video或image

“framePerSecond”:“-1“,//每秒抽取多少帧进行计算，-1为使用视频的原始帧率

}

输出格式：json

{

{“seconds”:”15.21“，//单位：秒，考虑到视频每秒帧数，结果精确到两位小数 }

}

## 单脚站立时身体摆动频率

输入

地址：http://IP:Port/ai/alphapose/swingfrequency/v1.0

Metohd：Post

参数：{

“inputType”:”video”，//video或image

“framePerSecond”:“-1”,//每秒抽取多少帧进行计算，-1为使用视频的原始帧率

}

输出格式：json

{

{“frequency”:“1.1”，//单位：次/秒，结果精确到一位小数 }

}