

基于数据驱动的数码变焦自动化测试脚本

基于数据驱动的数码变焦自动化测试脚本1

1.

1、需求分析：数码变焦测试用例，需要打开Potplayer播放器遍历测试用例表格中对6种分辨率格式分别使用HidTool进行缩放移动画面的测试操作，再通过分析串口数据判断测试结果，再将测试结果写回到对应测试用例的结果列中：
6种分辨率： MJPG 3840*2160 MJPG 1920*1080 MJPG 1280*720 H264 3840*2160 H264 1920*1080 H264 1280*7202

1.1. 2、代码实现：测试用例读取和结果写入：

a、用例读取：使用Pandas库实现对测试用例的测试点进行读取，拆分得到每行测试用例对应的分辨率，测试步长，测试步骤，用于后续执行测试

b、结果写入：使用Openpyxl库实现对测试用例的结果写入，根据每条测试用例执行完成后回传的行数和测试结果，每条测试用例测完自动写入结果2

1.1.1. 3、代码实现：Windows端工具控制：

a、Potplayer自动化控制：使用uiautomation库对Potplayer软件进行控制，包括打开软件、关闭软件、进入设备设置页面、指定分辨率切换、获取该设备支持的所有分辨率功能进行封装

b、HidTool自动化控制：使用uiautomation库对HidTool软件进行控制，包括打开软件、关闭软件、指定步长放大操作、指定步长缩小操作、复位操作、指定步长移动操作功能进行封装2

1. 1、需求分析：数码变焦测试用例，需要打开Potplayer播放器遍历测试用例表格中对6种分辨率格式分别使用HidTool进行缩放移动画面的测试操作，再通过分析串口数据判断测试结果，再将测试结果写回到对应测试用例的结果列中：

6种分辨率：

MJPG 3840*2160

MJPG 1920*1080

MJPG 1280*720

H264 3840*2160

H264 1920*1080

H264 1280*720

1.1.2、代码实现：测试用例读取和结果写入：

a、用例读取：使用Pandas库实现对测试用例的测试点进行读取，拆分得到每行测试用例对应的分辨率，测试步长，测试步骤，用于后续执行测试

b、结果写入：使用Openpyxl库实现对测试用例的结果写入，根据每条测试用例执行完成后回传的行数和测试结果，每条测试用例测完自动写入结果

1.1.1.3、代码实现：Windows端工具控制：

a、Potplayer自动化控制：使用uiautomation库对Potplayer软件进行控制，包括打开软件、关闭软件、进入设备设置页面、指定分辨率切换、获取该设备支持的所有分辨率功能进行封装

b、HidTool自动化控制：使用uiautomation库对HidTool软件进行控制，包括打开软件、关闭软件、指定步长放大操作、指定步长缩小操作、复位操作、指定步长移动操作功能进行封装

4、代码实现：串行端口控制数据接收：

通过PySerial库实现对串口的控制，包括获取当前电脑已连接COM端口、串口

初始化、串口输入账号密码获取超管权限、串口打开高级别log输出、串口获取input回传数据、串口退出超管权限功能封装

5、自动化测试流程：

a、初始化串口对象->获取串口超管权限->打开串口高级别log输出

b、逐行读取每条测试用例的测试点-

>将测试步骤、测试步长、测试类型、测试分辨率传入下一步

c、根据测试类型进行分流执行->错误值测试\放大测试\缩小测试：

0 - 错误值测试case - [1, '0,MJPG 3840×2160P 30(P 16:9),-1'] --

弹框“操作失败”关键字

1 - 放大测试case - [2, '1,MJPG 3840×2160P 30(P 16:9),5']

2 - 缩小测试case - [22, '2,MJPG 3840×2160P 30(P 16:9),40,-1']

d、通过串口获取每次执行操作后返回的数据：x（当前画面所在横坐标）

、y（当前画面所在纵坐标）、w（当前画面的宽度）、h（当前画面的高

度）->将这些值代入公式进行计算，判断该条测试用例的测试结果：

a、错误值：

无缩放状态->缩小1step

[1、弹框操作失败，2、zoom: 1, step 1]

无缩放状态->过度放大53step

[1、弹框操作失败，2、zoom: 0, step 53]

放大40step状态->过度缩小53step

[1、zoom: 0, step 40，2、x: 1280, y: 720, w: 1280, h: 720]

b、有效值：

放大->右移动画面1step:

放大5step

[1、zoom: 0, step 5，2、x: 160, y: 90, w: 3520, h: 1980，3、direction: 3, step 1，4、x: 192, y: 90]

(通过w、h、x、y来判断放大、缩小、移动的值是否正确)

所有分辨率的放大缩小移动都是以4K的进行的

1、放大的公式: $w=w1-step*64$, $h=h1-step*36$

2、缩小的公式: $w=w1+step*64$, $h=h1+step*36$

3、右移动的公式: $x=x1+step*32$, $y=y1$

放大40step->缩小->右移动画面1step:

缩小1step

[1、zoom: 0, step 40, 2、x: 1280, y: 720, w: 1280, h: 720, 3、zoom: 1, step 1, 4、x: 1248, y: 702, w: 1344, h: 756, 5、direction: 3, step 1, 6、x: 1280, y: 702]