

MuMu 摸点小助手说明书（兼摸点攻略）1.1 版本

前言

本程序旨在提高 MuMu 模拟器点位制作的精度及提高摸点效率。最初开发的原因是看到牢 D（B 站 ID：DDDDarkSide）在 S27 野外佩洛洛斯拉总力战中因为日向 13 环点位精度需求问题不得不用牙签戳屏幕来摸点。因此在参考了知屋（B 站 ID：知屋今天吃饭了吗）及皮革（B 站 ID：小容器不想努力了）的点位计算教程（视频 BV 号：BV1BLYCetEuX）和牢 K（B 站 ID：A_karin）、小源（B 站 ID：我心安处_便是皈依）提供的思路后（其中计算教程中的公式存在微小错误，小助手中已修正。）编写了这款软件。

在此鸣谢编写过程中测试并提出需求/改进意见的朋友们：不要啊老师（B 站 ID：黑暗大法师 DarkFaShi）、离殇语前辈（B 站 ID：神山识____）、郁香香（B 站 ID：寻棋问墨）、黎仙（B 站 ID：黎仙 orz）、Mirai（B 站 ID：MiraiDesu）、早坂爱（B 站 ID：早坂早坂愛）、原春幼（B 站 ID：林樱落的三雪三秋）、echo_nine（B 站 ID：我是很厉害厉害的哦）。

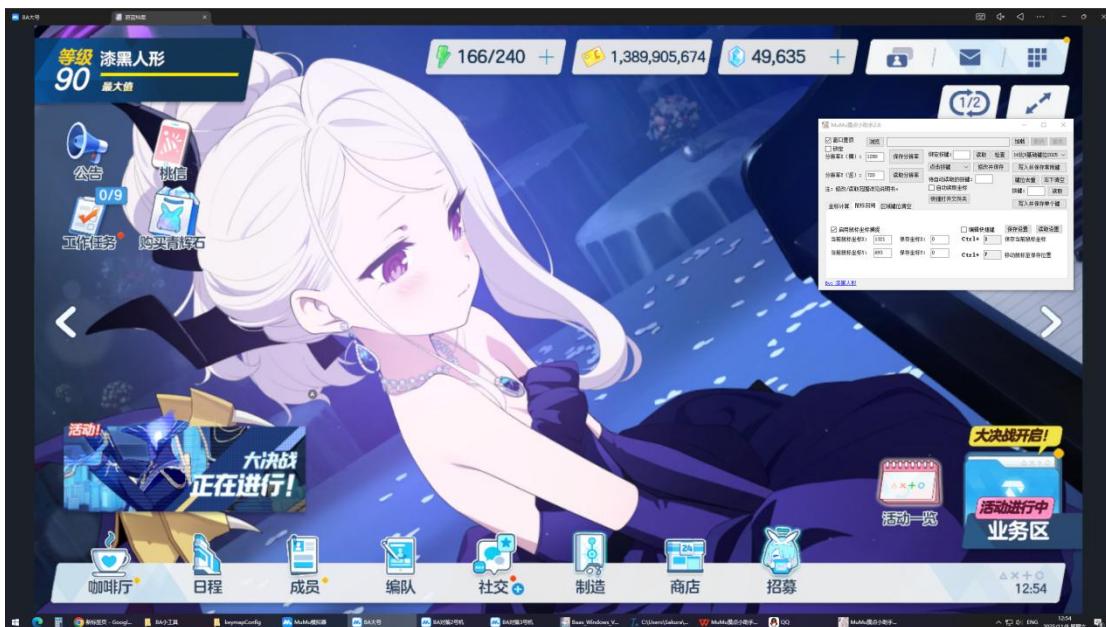
1、摸点教学/点位复制等常用功能介绍

好了，让我们直接开始摸点流程和常用功能的教学，繁琐的基础功能介绍留到后面的篇幅，有兴趣的小伙伴可以自行了解，开发一下你的脑洞，组合属于你的摸点方式。

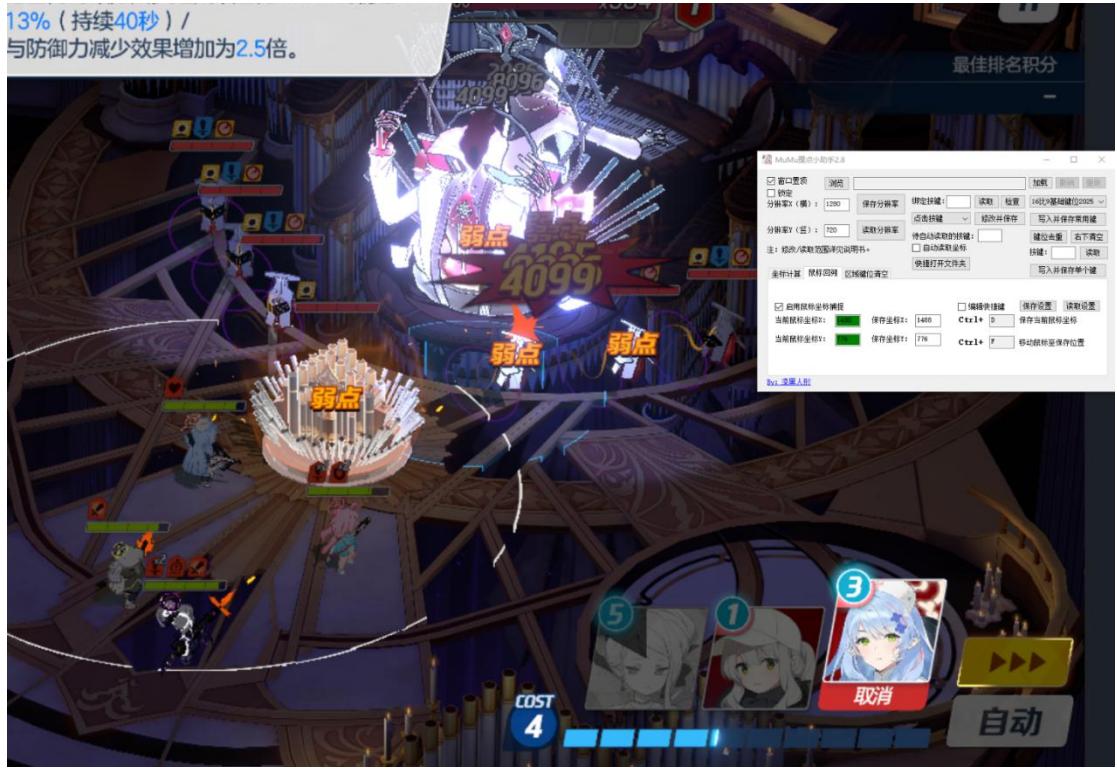
1.1、2.8 版本更新后适合懒人的精准摸点流程

1、首先，打开你的 MuMu 模拟器和摸点小助手，为了保证模拟器相对位置不改变，建议最大化模拟器窗口。然后勾选小助手的 窗口置顶 选项，保证小助手不被模拟器覆盖掉，底部功能模块切换到 鼠标回溯 区，勾选 启用鼠标坐标捕捉 将其功能打开。（快捷键默认为 **Ctrl+D** 和 **Ctrl+F**，该快捷键可修改成你喜欢的设置并保存。）

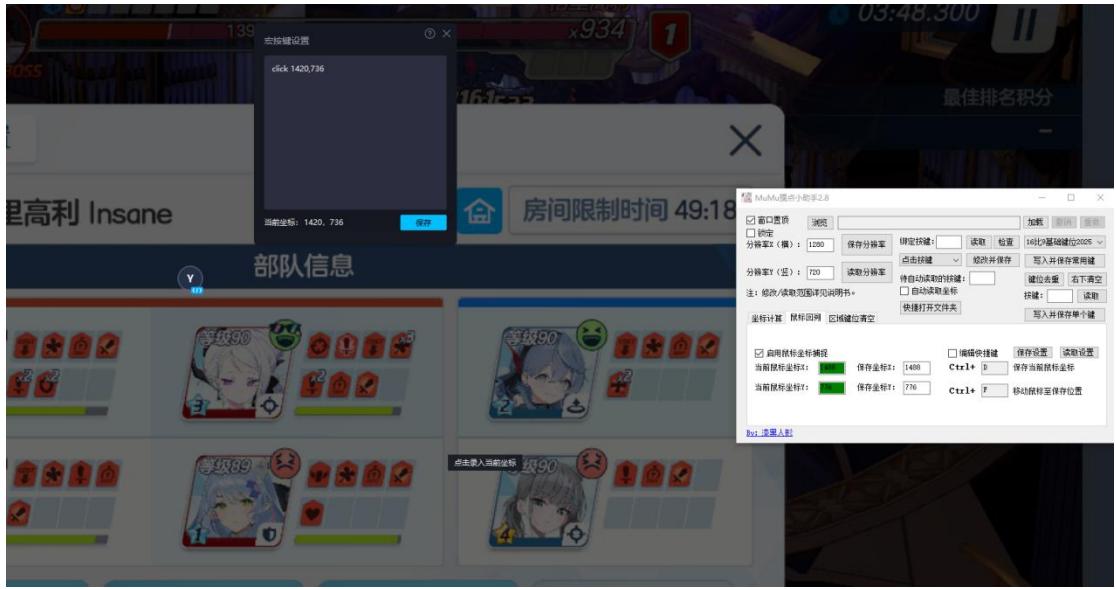
前置工作结束后完整桌面效果如图所示：



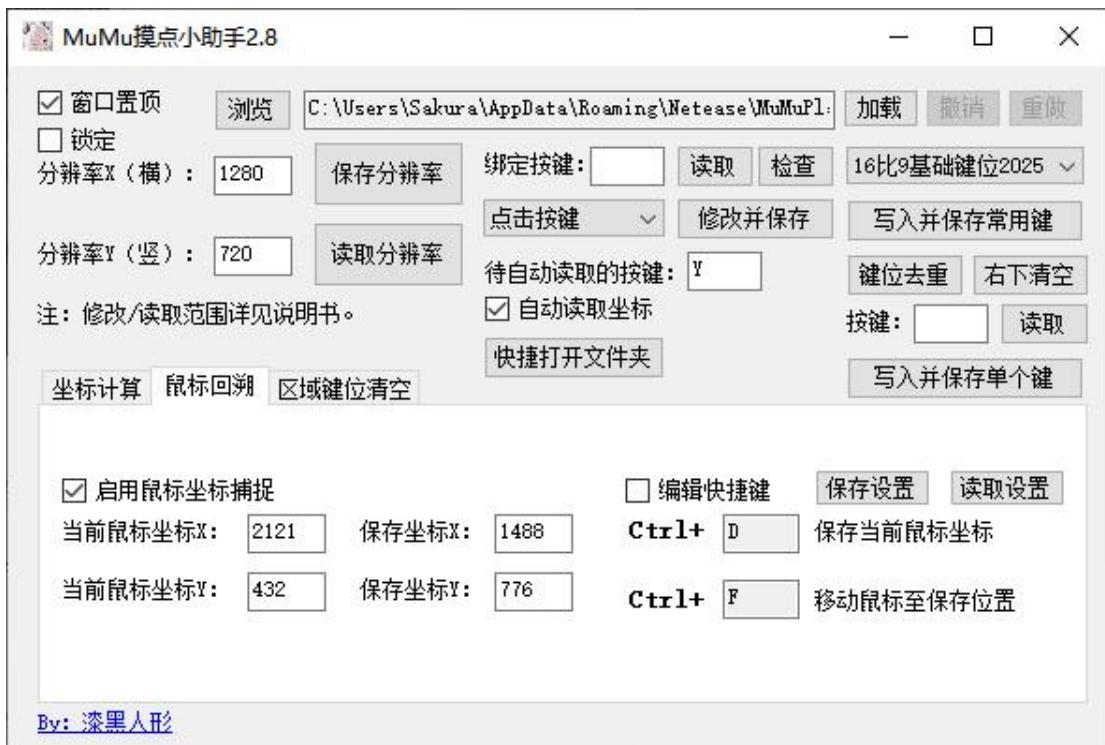
2、前置准备工作结束后，进入你需要摸点位的模式，按正常流程往下打，鼠标拖动角色技能到指定释放位置时，按一下 **Ctrl+D**，保存当前鼠标指针位置，效果如图（因截图功能限制，未能显示鼠标）：



3、保存好鼠标坐标后，暂停游戏，打开 MuMu 模拟器上方的键鼠方案编辑模块，新建或选择一个你已有的键位文件，进入编辑界面。右键新增一个用来读取坐标的工具键位，按键类型选择宏按键，设定一个不冲突的按键即可，这里设定为 Y 键。然后在宏按键的代码编辑部分输入 click，按 **Ctrl+F**，回溯鼠标坐标到先前的位置，确认鼠标位置无变化后单击一下，在 click 后记录坐标（如果代码编辑页面挡住了鼠标位置，可以先移开再继续操作。），完成操作后效果如图所示：



4、保存宏按键代码，保存按键文件，打开 MuMu 模拟器上方键鼠方案编辑模块，点击管理，进入管理页面并找到你刚刚创建或修改的键鼠方案，选中它，在更多操作中选择打开文件位置，找到该方案对应的 json 文件位置。（该操作在 V5.9.0 版本的 MuMu 模拟器中可用，旧版的需要先在操作录制中新建一个方案，随便录制点什么都行，然后点击文件夹图标，进入存放录制方案的 gameScript 文件夹，再进入它的上一级 data 文件夹，在 data 文件夹中找到 keymapConfig 文件夹，进入并按修改日期递减排序，即可快速找到最后修改的按键文件。详细操作参考我之前录制的教学视频（BV 号：BV1nM7nzJEcy）的 P4 部分。）找到键位文件后将其直接拖到小助手界面上，小助手会自动读取该键位文件，并将路径显示在上方。对应的文件夹路径将被记录，下次打开小助手时点击 **快捷打开文件夹** 按钮即可快速进入存放键位文件的文件夹。此时 在待自动读取的按键后的文本框 中输入工具人键位，演示中为 Y 键，勾选 **自动读取坐标** 小助手会自动读取 Y 键的坐标并显示在 **坐标计算** 区域中，到这一步后小助手界面如下图：



切换到坐标计算页面后可看到坐标已被成功读取(实战不用切):

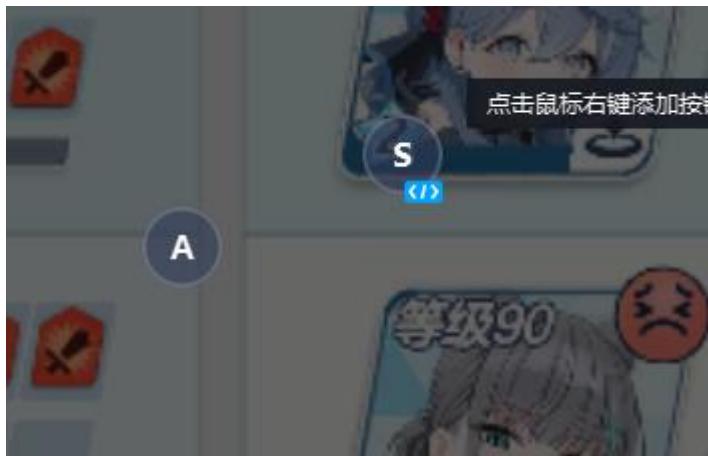


5、在 **绑定按键后的文本框** 中输入你要使用该坐标的键位，点击修改并保存，这里会分两种情况，如果已有该按键，会修改按键坐标到刚才读取的坐标中（目前仅支持普通点击按键及指定格式的宏指

牌按键)；如果没有按键则可以在旁边的下拉框中选择你要生成的按键(备注：该方式生成并写入后的按键有时会出现文字和 MuMu 模拟器正常生成的按键文字不一致的情况，但不改也不影响使用，在模拟器的按键编辑中重新保存一次即可恢复正常。)，默认是点击按键，也可根据需求生成宏指牌按键，小助手提示用户，确认后会生成相应设置的按键并写入键鼠方案的文件中。这里我们生成了一个点击按键 A 和宏指牌按键 S，(宏指牌的效果是鼠标放在你要开的牌上，按住该按键不放会最速宏开鼠标选择的牌到指定位置。)效果如图：



其中 S 键被 A 键隐藏到下面了(同坐标两个按键重叠的问题在 2.8.1 版本已修复，详见更新说明)，S 键是宏按键，可以随意拖动摆放到任意位置，因此我们进编辑页面，把 S 键拖出来即可看到我们已经成功生成了两个需要的按键：



之后我们可以以同样的手法，去摸其他的点位，此时小助手经过上面步骤的设置，会自动读取 Y 键的新坐标，我们只要更新 Y 键的坐标就能同步更新小助手里的坐标并写入其他键位中。简易流程为：捕捉鼠标坐标→进入编辑页面→删掉 Y 键宏代码中 click 后旧的坐标部分→回溯鼠标坐标，单击一次，录入新的坐标→保存宏按键代码→保存按键文件→通过小助手写入新坐标到指定按键或生成指定按键。**重点提示：**小助手的任何修改都可以通过右上角 撤销/重做 按钮（快捷键为 Ctrl+Z/Ctrl+Y）进行相应的操作。

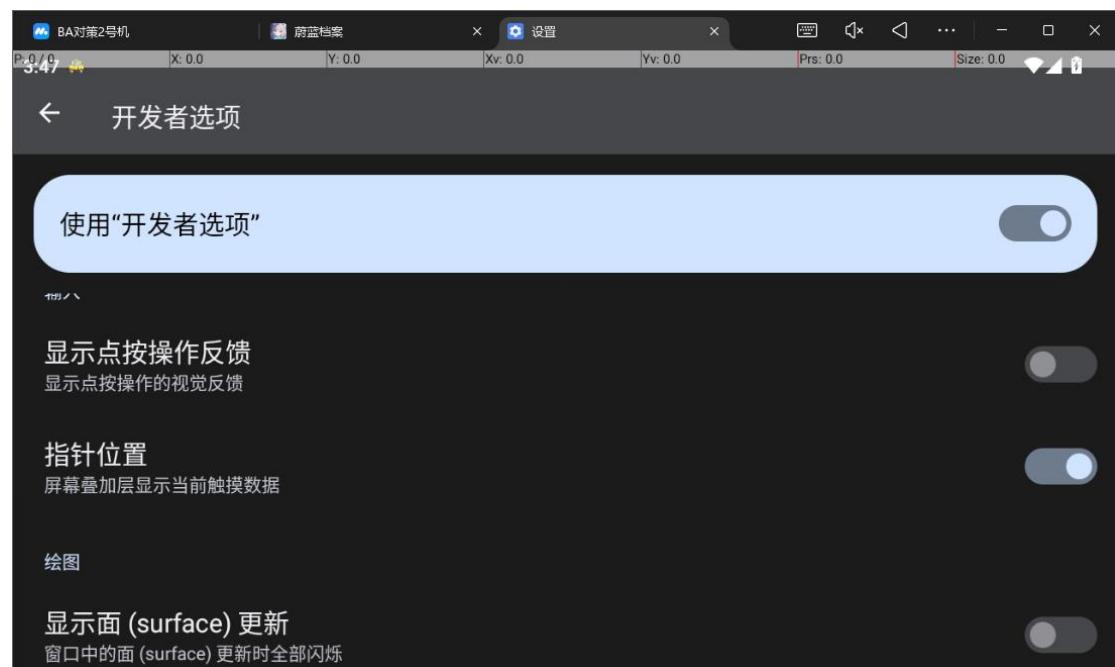
简单讲一下两种预设按键的特点，点击按键比较灵活，可以在释放之前按住看范围，如果不对的话，可以 ESC 连接两下取消释放并暂停游戏；宏指牌则通常用于卷分的场景，要开的牌不固定位置，但又需要最速宏开到指定位置的情况下，如鸡斯拉的最后一轮日向/黑兔收尾炸点、此前格里高利下 P 的渚，收尾的工头，大决战街蛇最速日鞠、最速斜开礼奈第一枪等等情景。

最后说一下该种摸点方式的优势：不用额外打开坐标轴，也不用设置分辨率这些东西，因为是直接读取 MuMu 模拟器生成的坐标，无需计算，所以精确度是 100%，你鼠标点在哪里，生成的就一定

哪里的点位。有点小缺点就是想要录像复盘的话需要打开 OBS 等第三方软件将模拟器和小助手一同录下，无法像之后的坐标轴方法一样用模拟器自带的录像即可进行复盘。

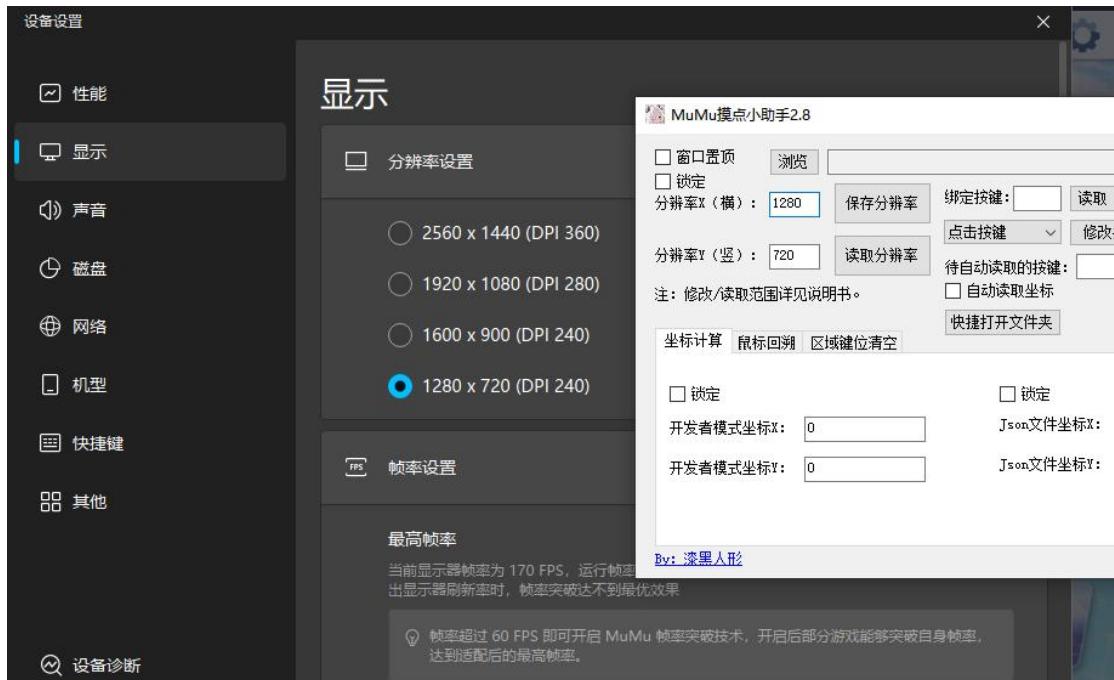
1.2、适合卷狗使用的通过内置坐标轴摸点

1、让我们先调整好 MuMu 模拟器内的设置，打开内部的坐标轴显示，参考我之前录制的教学视频（BV 号：BV1nM7nzJEcy）的 P1 部分，简单的流程为：打开设置→拉到最底部，点击关于手机→拉到最底部，连点 7 次版本号进入开发者模式→返回设置菜单，点击系统→滑到底部，点击开发者选项→下拉找到指针位置选项，打开。成功后屏幕上方会出现一条坐标轴，这也是有些 UP 主发轴的时候上方会带的东西，方便有心的观众自行按需求制作点位。效果如图：



2、之后调整小助手内的设置，点击 MuMu 模拟器上方的…，打开设备设置→点击显示，查看当前模拟器的分辨率，将对应的横坐标

X 和竖坐标 Y 填入小助手中。如后续修改了模拟器的分辨率请一定要记得同步修改小助手内的分辨率，不然点位计算会出错。点击 **保存分辨率** 后，下次打开小助手会自动加载你常用的这个分辨率。设置好后的效果如图所示：



3、该摸点方式可以任意调整模拟器大小到你习惯的位置，然后参考 1.1 内说过的內容，将你想要修改的键鼠方案文件拖入小助手中，小助手下方的功能模块中切换到 **坐标计算** 模块，然后进入你想要摸点位的模式，按正常流程往下打，鼠标拖动角色技能到指定释放位置时，按下截图键或直接全程录屏，反正我们要的是坐标轴显示的数据。当然如果记忆力足够好，不截图也没关系。总之我们把得到的坐标写入小助手中，小助手会自动根据分辨率计算 MuMu 键鼠方案文件中应写入的坐标数据。效果如下图所示：



4、得到坐标以后即可参考 1.1 的方式，直接写入坐标或者生成键位，同样也可以撤销修改，这里不再赘述。

5、为什么我会说这种方式适合卷狗呢，因为该摸点方式不影响你打轴的时候正常操作，只要开启录屏，事后从录像里找到正确的坐标即可。

1.3、案例分析：S2 大决战室外鸡斯拉的红甲睦月点位微调

既然讲到坐标计算这块了，那我们来一段案例参考，不感兴趣的可以跳下一段了。

我们先来一段聊天记录：







漆黑人形

这个我知道

漆黑人形

那是得往左边一点



Mirai

我昨天用小源的，调了5个像素

漆黑人形



这里大概是几个像素点

漆黑人形



黎仙orz

你要不然拿我的调一下吧



黎仙orz

分享《蔚蓝档案》按键操作方案，复制此分享码：
mumu9900435382，打开MuMu模拟器，在方案
管理中导入使用！
方案适用分辨率：
1920*1080
MuMu模拟器下载：<https://mu.mumu.163.com/>



Mirai

但是5个像素没看到右边是否正常

漆黑人形



我感觉这个位置容错很高啊

漆黑人形

再往左边一点呗

漆黑人形

这个圆清晰很多





这一段是当时 S2 大决战鸡斯拉红甲睦月点位的调整过程，由小源出手先手动摸了一个基础的点位，但小源手动存在误差，实战中发现偶尔有睦月炸不满的情况出现，大大降低了成刀率，于是参考大仙给出的炸歪部分的截图：



我们可以清晰的看到，小源的原始点位在右侧小鸡部分存在很大的冗余空间，而左侧则是擦边，这也是该原始点位少炸一只鸡的根本原因。因此在研究了图片之后，我通过摸点小助手读取了小源的点位坐标，并进行了微调，在左移三个像素点后成功得到了一个稳定的睦月满炸点位。

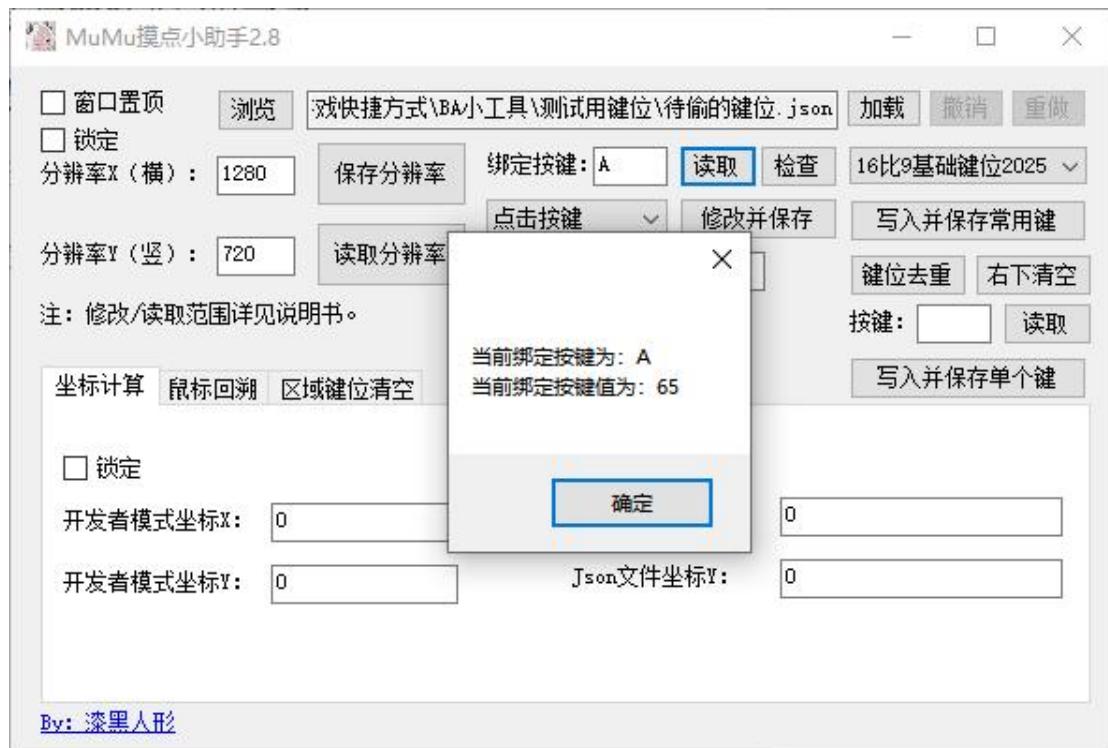
小助手中的坐标轴参数与系统相同，即，当模拟器分辨率为 1280x720 时，视左上角为(0, 0)，右下角为(1279, 719)，横纵坐标分别向右和向下延伸。因此我们横坐标向左移动 3 个像素即为横坐标-3，纵坐标向上移动 3 个像素即为纵坐标-3，相应坐标输入小助手后即可得到对应的键位文件内的坐标参数，并通过小助手写入相应按键。

你学会了吗？

1.4、点位/键位复制功能

该功能结合了坐标/键位读取和坐标/键位写入功能，有多种灵活的实现方式。（注意，小助手内的坐标读取功能仅支持单击按键和宏按键中的第一个坐标）有时候我们得到了一份别人做好的键位，但又更喜欢使用自己擅长的一套基础按键，如作者本人就更喜欢将别人的点位挪到自己惯用的 123 选牌，QWE 宏开牌，V 逐帧暂停，R 暂停界面一键重开……这么一套键位中。下面给出几种复制点位的参考实现方法：

1、坐标复制：先通过分享码或其他方式得到别人摸好的按键文件，拖入小助手中，然后你可以在如图的绑定按键这里输入你要读取的按键，然后点击读取，获取该按键坐标。如图：





点掉提示后即可读取坐标到计算区中，之后记得拖入你要写入的坐标文件，参考 1.1 的流程将按键写入即可。



当然你可以会说这样文件拖来拖去太麻烦了，我们可以看到小助手的压缩包解压后，根目录下会有一个 data 文件夹，我们可以将待

读取的按键文件复制到 **data** 文件夹中。如图，右侧整个模块都是为 **data** 文件夹中的 **json** 文件服务的：



放入文件后，点开下拉框，找到对应文件名的键位文件并选择：



同样在 **按键** 后的文本框中输入你要读取的按键，然后按下右侧 **读取** 键即可将坐标读到输入框中，该方法不需要切换文件，待写入的文件拖入后可以右侧读取按键，左侧写入按键。例：我们将待偷的键位中的 S 键写入测试空白 1 文件中的 A 键：



如图操作即可

2、键位复制：有时我们要偷的键位是一个宏按键，这时候就没办法用仅用偷坐标的方式去复制按键了。这时就只能将待偷的键位放入 data 文件夹中，然后输入你要偷的指定按键；将待写入的按键拖入小助手中，点击 写入并保存单个键 即可将该按键整体打包复制到小助手中。如果按键存在冲突，小助手会进行提示，可以强行写入，但写入后需要用户自行进入 MuMu 模拟器编辑，调整按键，冲突的按键会被模拟器标红，改成不冲突的其他键就好了。

1.5、预设基础键位注入功能

该功能的开发结合了键位读取、键位写入、键位删除等功能。该功能开发的原因是有时候需要偷的按键太多，不如我们反向将预设好的基础键位注入别人摸好的点位文件中。

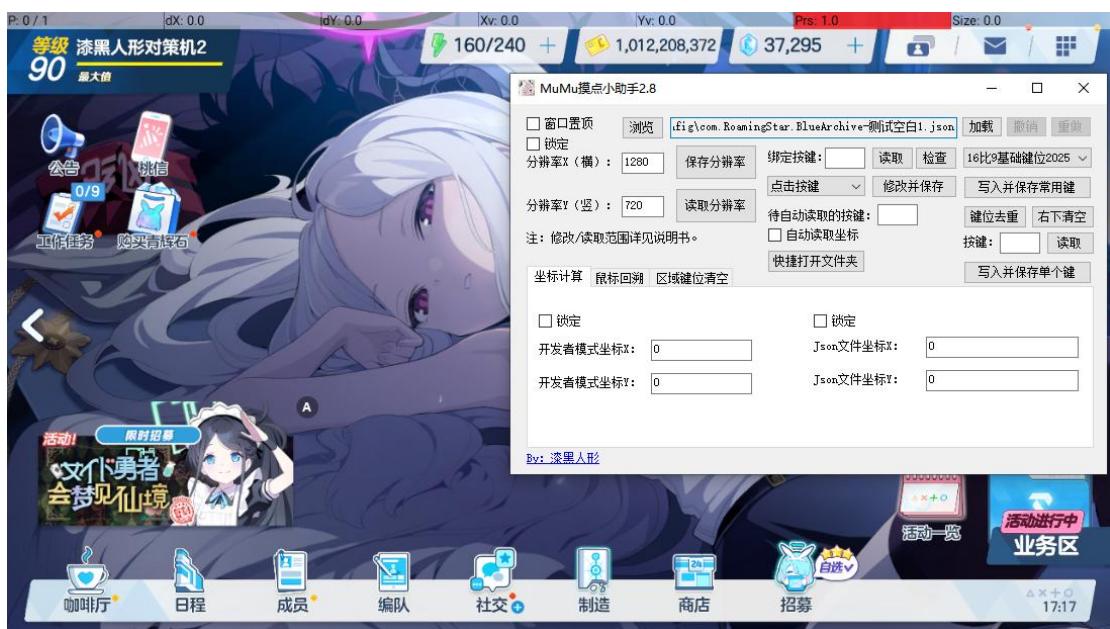
1、预设键位写入：小助手初始自带我常用的几款基础预设键位，用户也可以自行将预设键位导入 MuMu 模拟器修改成自己习惯的键位后再重新复制回 data 文件夹中，因为最好的编辑器就是 MuMu 模拟器本身。

当然有能力的用户也可以将其他自己习惯的预设键位复制到 data 文件夹，因为该功能的本质是进行重复按键检查后，将待写入的键鼠方案文件的按键部分整体写入另一个文件中，用户可以选择无视重复按键的警告强行写入，但后续需要自行修改冲突按键。

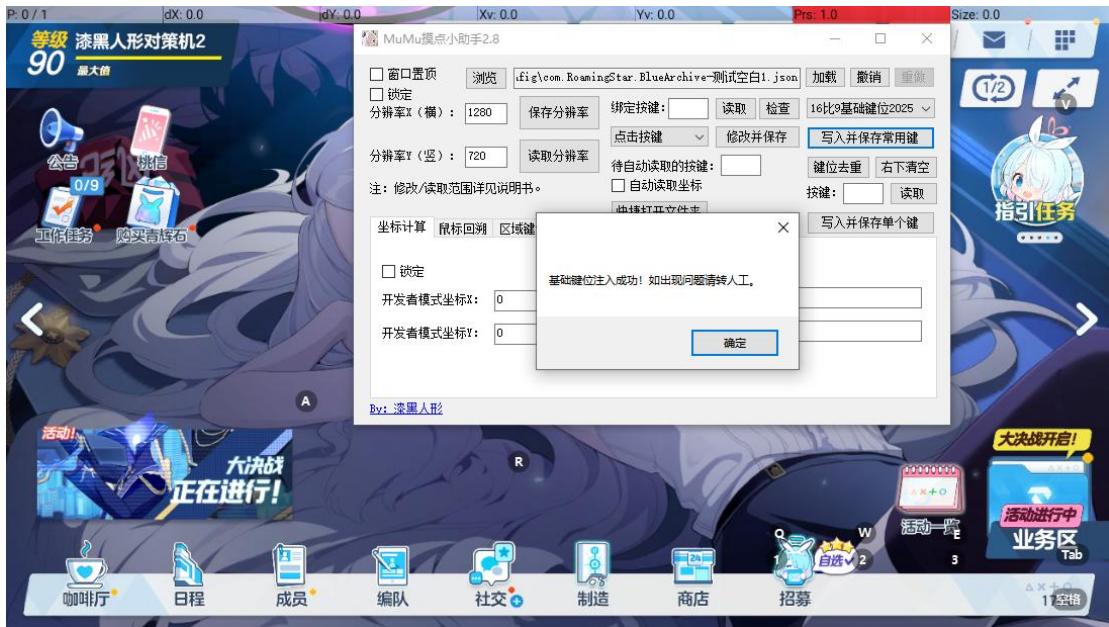
同样如图所示的模块：



导入待写入的按键文件但未写入的效果：



按下 写入并保存常用键 写入基础键位后的效果：



可以清楚的看到在我预设的 QWER123V Tab 空格等按键均被复制到了测试空白文件中。

3、去重/指定区域按键清空：有时我们拿到的按键文件可能与我们的基础预设文件存在冲突，直接写入时会提示存在按键冲突，此时点否，小助手会提示有哪些按键冲突，如图：



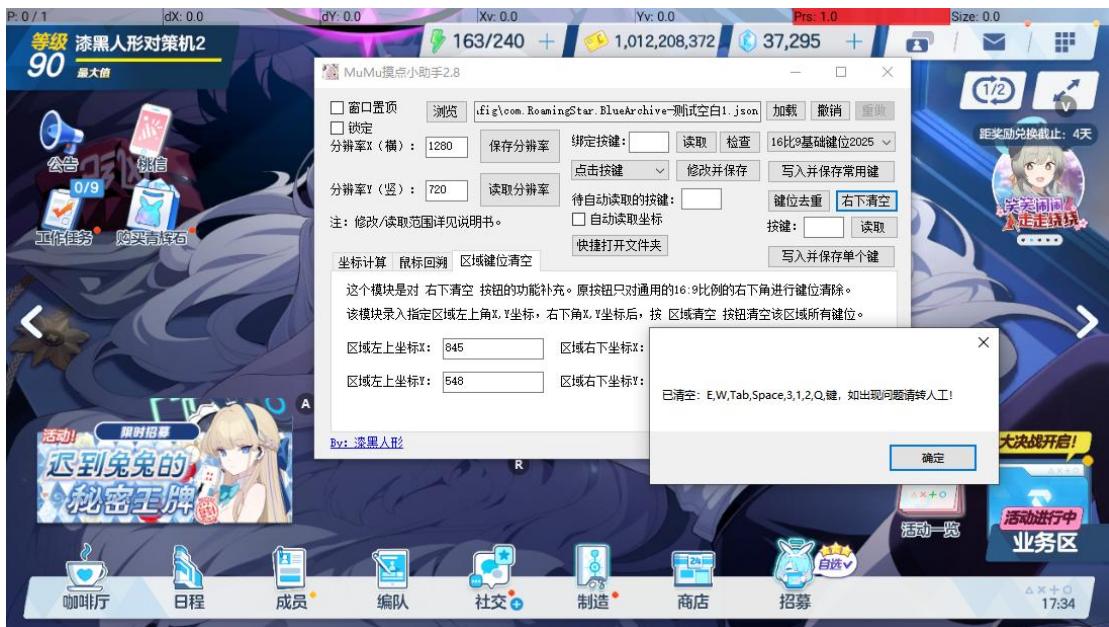
此时用户可自行检查，编辑待写入的按键文件，保留需要的部分，然后通过 **键位去重** 按钮简单粗暴的删掉所有重复键位，再重新进行基础键位的注入。

或者有时候我们拿到的按键文件虽然没有键位重复，但右下角存在分享者自带的常用键位，我们可以简单粗暴的用预设的 **右下清空**

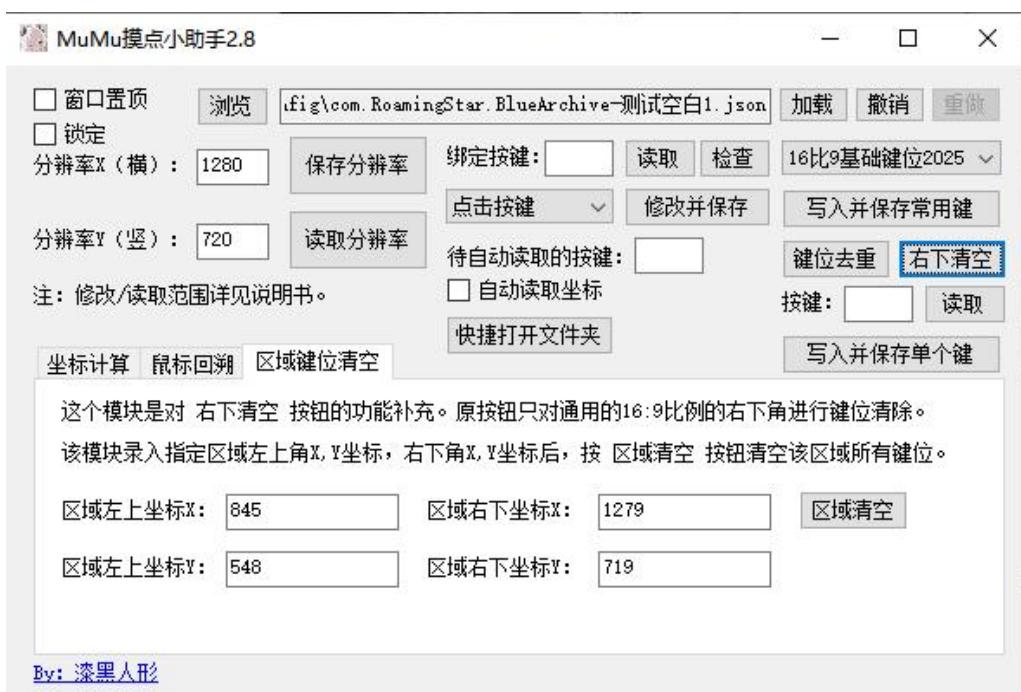
按钮(该按钮预设为清空 16:9 分辨率下右下角选牌区域的所有按键。)

来清空不要的键位，再写入自己的预设基础键位。

清空效果如图所示：



同样，模块 **区域键位清空** 属于对该功能的补充，通过左上角和右下角的坐标定位，来清空其他情形下指定区域的按键，下图为 1280x720 分辨率下对右下角选牌区的清空设置参考：

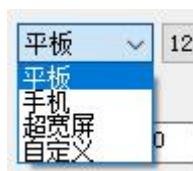


1.6、2.8.1 便利性更新详细说明

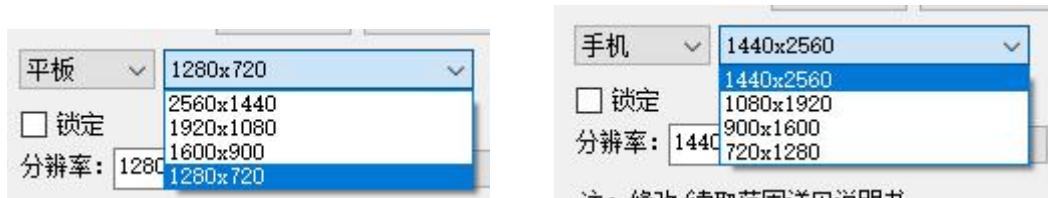
1、分辨率下拉框

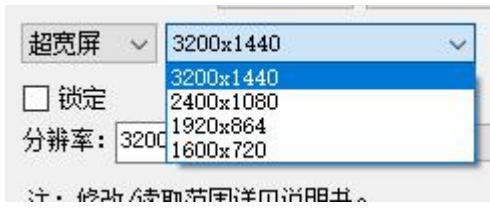


位于左侧，是对原本分辨率功能的一个补充，按 MuMu 模拟器的预设进行编写。左边的下拉框选择平板/手机/超宽屏/自定义等预设，如图：



切换后在左边可以选择对应的分辨率，如图：





选择的分辨率会直接修改到原本分辨率的输入框中，修改输入框中的分辨率也会尝试同步到下拉框中。如果是不存在预设的分辨率则切换到自定义一栏，未保存的分辨率会带*号，如图：



点击保存可将当前自定义分辨率保存进程序配置文件，支持保存多个自定义分辨率，同时在切换到自定义时会出现 **删除** 按钮，该按钮用于删除当前已保存的自定义分辨率。

下拉框支持方向键选择文件，按回车确定选择。

2、Json 文件夹下拉框

如图部分：



首先说明一下该下拉框对应的文件夹路径获取机制：

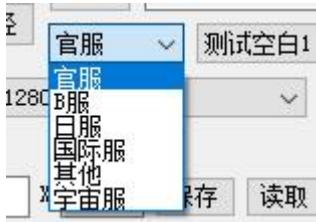
1、首次打开小助手时，自动根据环境变量获取当前用户名的用户文件夹再加"\AppData\Roaming\Netease\MuMuPlayer\data\keymapConfig"得到 MuMu 模拟器存放按键文件的文件夹路径。

2、手动通过 **重新获取路径** 按钮获取，机制相同。

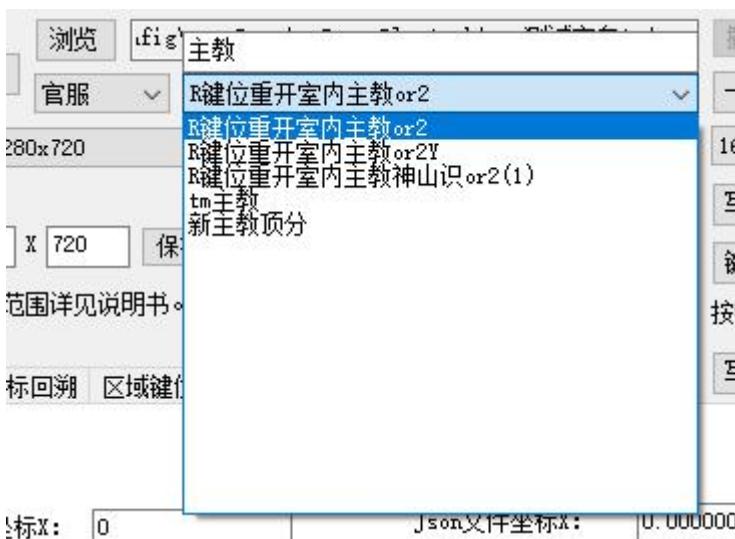
3、直接通过 **浏览** 按钮或者将 json 文件拖入小助手的形式，

加载当前文件并自动获取该文件所在的目录作为 json 文件夹。

通过以上方式得到文件夹路径后，在左侧下拉框会根据官服/B服/日服/国际服/其他 将 json 文件进行分类，分类依据为对应服务器游戏安装包的包名。通过右侧下拉框选择，宇宙服为彩蛋环节。



右侧下拉框点开后可以选择对应文件，如果找不到可以在搜索框中输入关键词，对文件进行进一步筛选。



注意：点开下拉框直接输入文字即可搜索，鼠标点击搜索框会导致下拉框关闭。

通过浏览按钮或者直接拖入的方式导入 json 文件时会自动根据分类切换对应的下拉框。且 json 文件夹会被更新为导入文件所在的文件夹。

右侧 打开 按钮可以打开 json 文件夹。

下拉框支持方向键选择文件，按回车确定选择。

3、Data 文件夹下拉框



显示小程序根目录下 Data 文件夹内存放的 json 文件，下拉框支持方向键选择文件，按回车确定选择。同样带有搜索框，用法和注意事项相同。

→导入↓ 按钮可将 json 文件下拉框中的文件导入到 Data 文件夹中，以去掉包名仅保留文件名的形式。 打开↓ 按钮打开根目录下的 Data 文件夹，删除按钮删除当前 Data 文件夹下拉框中选择的 json 文件。

4、强制替换按键

未勾选时与之前操作机制一致，勾选后修改按键时将删除原先按键，然后根据预设的点击/宏指牌按键，写入新的按键。如果默认单纯改坐标的修改方式出现问题可以勾选这个强制替换再进行修改。不过正常点的用法是做点位的时候用来偷懒，这功能是黎仙摸点的时候给我提的，我就加上了。

2、一些基础知识及小助手模块单独说明：

接下来的内容可能有些枯燥？感兴趣的可以看一下。

2.1、关于摸点的一些基础知识

1、坐标的计算部分

前言提到，知屋发布的视频中存在计算公式的错误，错误点在于模拟器内部的坐标轴是从(0,0)开始计算的，以 1280x720 的分辨率为例，内部坐标的范围是(0,0)~(1279,719)。因此 Json 文件内的实际坐标值的计算应是内部坐标/(分辨率-1)。如果没有摸点小助手的话这部分可以手算解决，反正也不难。

2、Json 文件里关于按键的格式

```
{
    "editor_icon_scale": 1,
    "icon": {
        "background_color": "00000066",
        "description": "",
        "radius_correction": 1,
        "rel_position": {
            "rel_x": 0.260359655981235,
            "rel_y": 0.617524339360223
        },
        "visibility": true
    },
    "key": {
        "device": "keyboard",
        "scan_code": 30,
        "text": "A",
        "virtual_key": 65
    },
    "rel_work_position": {
        "rel_x": 0.260359655981235,
        "rel_y": 0.617524339360223
    },
    "type": "Click"
}
```

如图所示，这是一个简单的点击按键的格式，大部分的设置我们不用管，因为改不到这一块，我简单说一下手动修改需要的部分：

"rel_position": 包含的"rel_x","rel_y"对应按键在模拟器中显示的位置。

"rel_work_position": 包含的"rel_x","rel_y"对应按键在模拟器中实际点击的位置。

"scan_code": 对应按键的扫描码。

"virtual_key": 对应按键的键值

"text": 这个不用我说，一看就明白，一般手动搜索并修改就是通过这个来定位按键在 json 文件里的具体位置。

手动修改点击按键的坐标可以只修改"rel_work_position"后的 X,Y 坐标，将对应 X,Y 的计算结果替换进去即可，保留的小数位数越多精度越高，但也不用保留太多。这种修改方式保存后你会发现模拟器中的对应按键呈现一种错位效果，即按键实际点击的位置和按键所在的位置不一致，这时候点开 MuMu 模拟器的编辑页面，什么都不用改，重新保存一次，模拟器会自动修正"rel_position"中的参数与"rel_work_position"同步。

而在宏按键中，新增了"press_actions"和"release_actions"用来存放按下和释放时触发的代码，如图，这是一个简单的宏按键格式及其代码部分在编辑器中的表现形式：

```
{  
    "editor_icon_scale": 1,  
    "icon": {  
        "background_color": "00000066",  
        "description": "",  
        "radius_correction": 1,  
        "rel_position": {  
            "rel_x": 0.25346579218106996,  
            "rel_y": 0.2148554850604631  
        },  
        "visibility": true  
    },  
    "key": {  
        "device": "keyboard",  
        "scan_code": 19,  
        "text": "R",  
        "virtual_key": 82  
    },  
    "press_actions": [  
        "sleep:100",  
        "click_rel:(0.435391, 0.701318)",  
        "sleep:100",  
        "start_loop:10",  
        "sleep:50",  
        "click_rel:(0.510288, 0.648609)",  
        "stop_loop"  
    ],  
    "rel_work_position": {  
        "rel_x": 0.25346579218106996,  
        "rel_y": 0.2148554850604631  
    },  
    "release_actions": [  
    ],  
    "type": "Macro"  
},
```



但实际我们在模拟器中编辑的代码只会放在"press_actions"模块中，需要修改的话则要对照模拟器中编辑的代码来修改对应的坐标，比如图中的(0.435391,0.701318)对应的坐标就是click 444,403。

而小助手中实际代码的实现则是通过"virtual_key"中的键值来定位按键在 json 文件中的位置，再进行后续操作。这部分感兴趣的可以直接去看源代码。

3、关于宏按键中的代码

请不要直接在编辑界面复制带有坐标的宏按键代码给别人使用，因为，如上图中的案例，click 444,403 后面跟随的这组坐标跟模拟器窗口被拉伸、缩放的大小比例有关的。当模拟器更换分辨率并重启之后未手动缩放窗口时，这组坐标与开发者模式下的坐标显示相同。因此如果你想把你写的带坐标的宏按键分享给你

的朋友，建议直接分享按键文件，然后在同一个模拟器下复制代码，这样才能保证对应的坐标准确无误。

4、关于小助手根目录下 `data` 文件夹的美观问题

之前接到过 echo_nine (B 站 ID: 我是很厉害厉害的哦) 的反馈，考虑到一些人习惯将小助手直接解压在桌面，而该版本自带的 `data` 文件夹某种意义上可能会不太美观？或者说不够简洁？如果有这方面顾虑的用户可以将小助手移到桌面之外的文件夹里，然后复制小助手的程序本体，在桌面粘贴时选择粘贴其快捷方式即可。

5、模拟器分辨率对按键文件的影响：

同分辨率比例的情况下，例大伙常用的 `1920x1080` 和 `1280x720` 均为 `16: 9`，这种情况下按键文件基本通用，位置上只有极为微小的偏差，除了极个别情况如黑白 P2 的安定点跳点会出问题，必须严格按照相同分辨率使用点位文件外，基本同比例分辨率的按键文件是可以通用的。但不同比例的分辨率之间的按键文件就需要自行调整了，一般来说无法通用。

2.2、摸点小助手的功能单独说明

1、窗口置顶：顾名思义，使用中如有需求可以勾选，使小助手的窗口不会被其他窗口覆盖掉。

窗口置顶

2、锁定：顾名思义，勾选之后对应部分的文本框会被锁定，无

法输入。这是初始版本留下的设计，因为那个版本没有坐标写入的功能，小助手只是纯粹的计算工具，这个功能一般用于锁定 json 文件坐标的计算结果，防止有人手滑按错。

锁定

Json文件坐标X:

Json文件坐标Y:

3、浏览/加载

这两个其实是装模作样做一下的功能，点一下能打开对话框让你手动选择文件夹，再手动选择需要导入的 json 文件，最后手动点一下加载来导入。其实连我自己也不用，直接把文件拖进小助手就自动加载了多方便，对吧。



4、撤销/重做

顾名思义，2.8 版本新增的功能之一，同时也设计了快捷键，和其他软件设计相同，撤销为 **ctrl+z**，重做为 **ctrl+y**。每次执行修改时小助手都会进行相应的备份，如果修改结果出现问题，最近可以撤销至文件刚导入时的状态。但要注意，如果撤销后手动进行另一次修改的话，将无法再进行重做，视为另一个分支。



5、分辨率设置区域

可对分辨率进行修改，并保存，保存后下次打开小助手则会自动加载上一次设定的分辨率。读取则是防止有人改错，未保存的

话可以手动读取上一次设定分辨率的记录。修改分辨率会同时进行 json 文件坐标的计算。



6、围绕绑定按键的编辑区

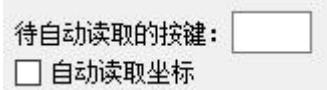
顾名思义，在文本框里按一下按键即可自动读取对应按键的信息，围绕这个按键进行读取/检查/修改并保存等操作。这部分操作对应的是拖入小助手中，路径显示上方长文本框中的文件。读取部分仅支持点击按键的坐标和宏按键的第一个坐标，读取后保存在下方坐标计算区域中的 Json 文件坐标对应文本框中。检查按钮则用于判断该按键的类型为点击按键或宏按键。其实本来应该是调试时候用来测试功能的按键，但放那挺和谐的，多少也有点用，后面就没删。下拉框在文件中不存在该按键时生效，选择指定的按键类型，根据参数生成对应的按键代码并写入 Json 文件中。修改并保存按钮则分两种情况，当文件中存在该按键时，如果是点击按键或指定格式的宏指牌按键则写入坐标（非指定格式的宏指牌按键则有概率将文件写坏，建议出问题直接撤销操作。），不存在该按键时则根据用户的设置生成一个对应按键并写入文件。



7、围绕待自动读取的按键的编辑区

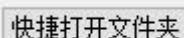
同上，该区域设计本意是实现 1.1 中的懒人摸点，勾选自动读取坐标后，工具人宏按键的坐标可以在每次按键文件被修改时实

时读取，勾选后更改要读取的按键同样会自动读取新按键的坐标。读取的坐标会被放在下方坐标计算区域中的 Json 文件坐标对应文本框中。



8、快捷打开文件夹

顾名思义，该按钮对应的功能为打开用户上一次打开的按键文件所在的文件夹。即为上方长文本框中对应路径指向的文件所在的文件夹。具体逻辑为用户通过浏览/手动拖入的方式打开文件后，自动存储对应文件夹的路径。用户下一次打开摸点小助手即可通过该按钮快捷进入存放按键文件的文件夹，不必通过 MuMu 模拟器来打开。



9、基础键位注入/点位复制区

该部分下拉框中文件会随着每次打开自动刷新，对应文件为小助手程序根目录下的 **data** 文件夹。将新的 Json 文件放入 **data** 文件夹中，重新打开下拉框即可看到对应的文件出现在其中。

2.8 版本的小助手在 **data** 文件夹中自带四个基础键位文件，分别为对应 **16: 9** 比例的分辨率下的常规选牌区三张牌的宏开键位和爬塔模式下五张牌的爬塔键位，以及超宽屏分辨率 **2400x1080** 下的常规模式/爬塔模式键位，按文件名来区分。

该功能区的原始设计意图为让用户自行编辑属于自己常用的键位后，存入 **data** 文件夹，之后可以通过写入并保存常用键按钮

逆向写入他人的点位中。使用时需要将待写入的文件拖入小助手，路径会自动读取并显示在程序正上方长条文本框中。写入时会检查按键重复情况，可强行写入，但写入后需要打开模拟器的编辑页面对按键文件手动进行编辑。

附带功能为支持其功能而编写的键位去重和右下清空。键位去重顾名思义，一键删除待写入按键文件中与基础键位重复的按键，使用时需要用户自行校验，不要误删需要的按键，如果发现删错请自行撤回。

右下清空为傻瓜式设计，默认为清空 16: 9 分辨率下右下角选牌区的所有按键，这个比例的分辨率最常用所以做成一键式的。联动扩展区域为区域键位清空部分，是对右下清空功能的一个补充，方便用户自行选中区域进行清空。

扩展功能为对某个按键的整体复制，用户可在按键：后的文本框中输入按键。之后按下读取按钮即为读取该按键的坐标，逻辑判定与其他读取功能一样，点击按键读取坐标，宏按键读取第一个坐标，并存入坐标计算区域中的 Json 文件坐标对应文本框中。

写入并保存单个键则为读取下拉框选中 Json 文件中的指定按键（通过按键：后的文本框进行指定。）完整写入待修改的 Json 文件中，遇到重复按键会提示用户是否继续写入，可强行写入，写入后同样需要在模拟器的编辑器中手动修正按键。参考本说明书 1.5 中的操作流程。



10、坐标计算模块

其实应该算是小助手的核心模块？不过确实用起来很简单，上方设置好分辨率后，下方在开发者模式对应坐标 X,Y 中输入坐标，即可在 Json 文件坐标中得到计算结果，该计算结果可以用多种方式写入你需要的地方。同时其他读取相关功能得到的坐标也会存在 Json 文件坐标的两个文本框中。



11、鼠标回溯模块

懒人摸点相关的核心模块，灵感来源于小源（B 站 ID：我心安处_便是皈依）提过的通过按键精灵的按键抓抓功能进行摸点的方式。但按键抓抓我也用过，本质上并非为了摸点而设计，有很多功能是多余的，摸点的时候用不到且会占用额外的资源。因此简化后提取我们需要的对鼠标坐标的显示功能，以及追加了一个回溯鼠标指针到相应位置的功能，就成了这个模块。

该模块需要勾选启用鼠标坐标捕捉来使用，默认全局快捷键为

Ctrl+D 保存鼠标坐标, **Ctrl+F** 回溯鼠标坐标, 该快捷键可在勾选编辑快捷键后进行编辑, 同样编辑后可保存/读取设置, 保存后的设置在下一次打开小助手时依然沿用。



12、By: 漆黑人形

没什么特别的意义，只是个链接，点了之后可以跳转到作者的B站空间，win10 系统下非管理员权限运行的话这个跳转还会被拦截。

[By: 漆黑人形](#)

13、2.8.1 新功能更新部分参考 1.6

3、2.8 版本之后的更新日志

3.1、2.8.1 版本, bug 修复兼便利性优化

新旧版本同功能但界面变化部分对照，以相同颜色的方框为准：



1、BUG 修复：按键删除功能新增对点击/宏按键之外其他按键的支持，现在支持大部分可以通过键值定位的按键了，少部分无法定位的按键 BA 玩家用不到，就没有单独适配。（BUG 说明：录制视频教

程的时候发现柚子厨给的点位文件删除连击按键 C 键的时候产生 BUG 导致修改后的内部格式出错，MuMu 模拟器无法读取。）

2、BUG 修复：重置了对点击按键坐标修改的代码，如果单纯修改还是出现问题，建议勾选强制替换按键，勾选后代码执行逻辑为先删除原本按键再根据固定格式写入新按键。（BUG 说明：此前的旧代码偶有出现修改坐标后少一个换行的问题，导致文件内部格式出现问题，MuMu 模拟器无法读取。）

3、BUG 修复：生成宏指牌按键时添加了对按键显示位置的坐标偏移（不影响宏实际点击位置）。（BUG 说明：此前的旧代码同坐标的点击按键和宏指牌按键生成在同一个位置，不好单独将宏指牌按键拖出来，黎仙提了这个问题之后我就给宏指牌的生成位置加了一点偏移，方便同坐标的两种按键生成之后可以区分出来。）

4、便利性优化：按左中右顺序，新增了分辨率，MuMu 模拟器 keymapConfig 文件夹中文件，小助手自带 Data 文件夹中文件三个下拉选项框。分辨率下拉框根据 MuMu 模拟器的设置预设了平板、手机、超宽屏三个，同时也支持用户保存/删除自定义分辨率。keymapConfig 文件夹下拉框根据包名分类 keymapConfig 中的 json 文件，带一个搜索框。Data 文件夹下拉框显示 Data 文件夹下所有 json 文件，同样带一个搜索框。

5、便利性优化：程序首次打开时会自动尝试匹配当前用户名下 MuMu 模拟器的 keymapConfig 文件夹，可通过 重新获取路径 按钮手动获取。

6、便利性优化：新增 强制替换按键 选项，勾选后修改并保存按键时会删除原本的按键，根据点击按键/宏指牌按键的不同选项重新写入新的对应按键。

7、便利性优化：原本的 快捷打开文件夹 按钮移至右上角 打开按钮，功能为打开上方地址栏中存放按键文件的文件夹，（一般自动获取的默认文件夹为模拟器的 `keymapConfig` 文件夹），新增 打开↓按钮，功能为打开程序根目录下的 `Data` 文件夹。

8、便利性优化：新增 → 导入 ↓ 按钮，功能为将左侧 `keymapConfig` 下拉框选中的按键文件导入程序根目录下 `Data` 文件夹，对应的，新增 删除 按钮，功能为删除 `Data` 文件夹下拉框中选中的文件。

9、体验优化：取消了一些导入/修改中不必要的提示，使操作更加流畅；优化了对文件被其他程序修改的监测提醒，现在不会重复多次提醒了。

后记：

先写到这里吧，暂时想不到还能补充什么内容了。截止目前小助手其实还有一些他人提供的有关便利性优化的建议和思路没有实现，这个就留着有空的时候再尝试做出来吧。毕竟我也不是什么很厉害很厉害的程序员，只是兴趣使然做了这么一个简单的小工具。能帮助到有需要的人，我就很高兴了。希望今后不会有人因为 MuMu 原生的

按键编辑功能不够精准而感到烦恼。

整活：族谱部分

附录：小团伙族谱挂名（整活向）：小容器的操控者 Sammy（B 站 ID: Sammywzx）、被抽陀螺又爱改名的星露土叔叔（B 站 ID: 夢醒时分 choral）、主包的情敌&超级日奈厨光吟（B 站 ID: 光光光怜）、旮旯给木高手组：对话会弹选项框的小孩空空瑠（B 站 ID: 空空瑠_），很专业很专业的三一柚子厨（B 站 ID: Kalaseba），很专业很专业的大手青空（B 站 ID: 私聞いて然）、旮旯给木&专栏双领域大手子迷梦（B 站 ID: 期待有趣的梦）、唱歌很好听跟主包对唱许嵩歌曲的芳狗（B 站 ID: 雏衣波莱特）、尊贵的小助手内测用户哈基 a（B 站 ID: UZquin）、神秘的一心杀手千早爱音厨（B 站 ID: 千早爱音厨）、沉迷大乱斗的肃夜（B 站 ID: pku 大乱斗玩家）、天天 4=1 骗人进大乱斗（这个肃夜也干了）的批火龙（B 站 ID: 东下悠然）、欠主包一个国一的足控财布（B 站 ID: 不是财布）、到处旅游的 CS 高手爱丽丝饱饱（B 站 ID: 爱丽丝好有感觉）、很帅很帅的违神（B 站 ID: 違命候）、前 B 服村长枪男 kke（B 站 ID: kke888）、三角洲高手组：很会改枪的人形 V10 久伴烟雨（B 站 ID: 久伴烟雨）、过了保质期的美少女 JK 凛爷（B 站 ID: 凛夜い yoruRin_），汲取尸体精华的百绊爱厨先空后白（B 站 ID: 先空后白）、日常小组招人的礼花肘（B 站 ID: 立花奏 01）（广告时间：小组 ID42 招人，广告词暂空，礼花肘没结

广告费。）、超级百绊爱厨牢爱（B 站 ID：爱丽丝 kiss）、和 jbyy 一样神秘的游戏爷（B 站 ID：Xx xX）、燃尽了的帕鲁（B 站 ID：空泷泽明）、劲大的吓人的 hyl（B 站 ID：化炎凉）、同样打红甲劲大的吓人的虹夏（B 站 ID：“竜華キサキ”）、已经成为尸体的考批 Soyo（B 站 ID：紫荆 7x）、在打复活赛的晕捂脸（B 站 ID：矢量空间）、上不了群一的悲情英雄 Sonya（B 站 ID：Sonya 嘴嘴）、小源的（屏蔽词）（B 站 ID：枫落 ouo）、智斗巅峰之祥爷假死（B 站 ID：原桃玩家）。以上排名不分先后，我去，这族谱比我命还长。