

一、接口简介

1. 通信基于 websocket 协议，端口默认情况下是 38999，可以通过配置文件修改。
2. 消息为 json 字符串格式，分为请求消息、响应消息和事件消息，通过 func 字段标识具体的功能。
3. 请求消息可以携带的字段除了 func 外，还可以有 iden 和该请求特有的字段。iden 可以用于标识该请求消息。
4. 响应消息可以携带的字段除了 func 外，还必须有 ret 字段（表示返回值），ret 不为 0 时必须有 err_info 字段（表示错误信息），可以有 iden 和该响应特有的字段。iden 与对应的请求消息的 iden 相同。
5. 事件消息可以携带的字段除了 func 外，还可以有 iden 和该事件特有的字段。iden 与对应的请求消息的 iden 相同。

二、基本接口

1. 设置全局配置

请求:

```
{  
    "func":"set_global_config",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    // 文件保存  
    "file_save_path":"C:\\\", // 默认是文档目录  
    "file_name_prefix":"Huago", // 文件名前缀  
    "file_name_mode":"date_time", // 可以是 date_time、random、  
    sn_date_time  
    // 图像保存  
    "image_format":"jpg",      // 格式，支持 jpg、bmp、png、tif、pdf、  
    ofd、ocr-pdf 和 ocr-ofd  
    "image_jpeg_quality":80, // jpg 质量，0-100  
    "image_tiff_compression":"none", // 可以为 none、lzw 和 jpeg  
    "image_tiff_jpeg_quality":80      // tiff-jpeg 质量，0-100  
}
```

响应:

```
{  
    "func":"set_global_config",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

2. 获取全局配置

请求:

```
{  
    "func":"get_global_config",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}  
响应:  
{  
    "func":"get_global_config",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    // 文件保存  
    "file_save_path":"C:\\\", // 默认是文档目录  
    "file_name_prefix":"Huago", // 文件名前缀  
    "file_name_mode":"date_time", // 可以是 date_time、random  
    // 图像保存  
    "image_format":"jpg",      // 格式, 支持 jpg、bmp、png、tif、pdf、  
    ofd、ocr-pdf 和 ocr-ofd  
    "image_jpeg_quality":80, // jpg 质量, 0-100  
    "image_tiff_compression":"lzw", // 可以为 none、lzw 和 jpeg  
    "image_tiff_jpeg_quality":80 // tiff-jpeg 质量, 0-100  
}
```

3. 加载本地图像

请求:

```
{  
    "func":"load_local_image",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_path":"C:\\1.jpg" // 图片本地路径  
}  
响应:  
{  
    "func":"load_local_image",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    "image_base64":"xxx" // base64 字符串, 带前缀  
}
```

4. 保存本地图像

请求:

```
{
```

```
        "func":"save_local_image",
        "iden":"56fc",      // 可以是随机字符串，该字段可以省略
        "image_base64":"xxx" // base64 字符串，带前缀
    }
响应:
{
    "func":"save_local_image",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\1.jpg"
}
```

5. 删除本地文件（为了安全，仅能删除后端生成的文件）

请求:

```
{
    "func":"delete_local_file",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "file_path":"C:\\1.jpg" // 本地路径
}
响应:
{
    "func":"delete_local_file",
    "iden":"56fc",
    "ret":0,
    "err_info":""
}
```

6. 清理全局文件保存目录（为了安全，只会删除后端生成的文件）

请求:

```
{
    "func":"clear_global_file_save_path",
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略
}
响应:
{
    "func":"clear_global_file_save_path",
    "iden":"56fc",
    "ret":0,
    "err_info":""
}
```

7. 上传本地文件

请求:

```
{  
    "func":"upload_local_file",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "file_path":"C:\\1.jpg",  
    "remote_file_path":"/path/1.jpg",    // 远程文件路径  
    "upload_mode":"http",  // http 或 ftp  
    "http_host":"192.168.1.100",  
    "http_port":80,  
    "http_path":"/upload.php",  
    "ftp_user":"xxx",  
    "ftp_password":"xxx",  
    "ftp_host":"192.168.1.100",  
    "ftp_port":21  
}
```

响应:

```
{  
    "func":"upload_local_file",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

8. 合成本地图像

请求:

```
{  
    "func":"merge_local_image",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_path_list":["C:\\1.jpg", "C:\\2.jpg"],  
    "mode":"horz",     // 可以是 horz 和 vert  
    "align":"top",       // 可以是 top, bottom, center, left 和 right  
    "interval":0,        // 间隔  
    "local_save":true,   // 是否保存到本地，默认保存  
    "get_base64":false  // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func":"merge_local_image",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

```
    "image_path":"C:\\3.jpg",
    "image_base64":"xxx" // base64 字符串, 带前缀
}
```

9. 本地合成多张图像

请求:

```
{
    "func":"local_make_multi_image",
    "iden":"56fc", // 可以是随机字符串, 该字段可以省略
    "image_path_list":["C:\\1.jpg", "C:\\2.jpg"],
    "format":"tif", // 格式, 支持 tif、pdf 和 odf
    "tiff_compression":"none", // 可以为 none、lzw 和 jpeg
    "tiff_jpeg_quality":80, // tiff-jpeg 质量
    "local_save":true, // 是否保存到本地, 默认保存
    "get_base64":false // 是否获取 base64, 默认不获取
}
```

响应:

```
{
    "func":"local_make_multi_image",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\3.jpg",
    "image_base64":"xxx" // base64 字符串, 带前缀
}
```

10. 拆分本地图像

请求:

```
{
    "func":"split_local_image",
    "iden":"56fc", // 可以是随机字符串, 该字段可以省略
    "image_path":"C:\\1.jpg",
    "mode":"horz", // horz,vert
    "location":1000,
    "local_save":true, // 是否保存到本地, 默认保存
    "get_base64":false // 是否获取 base64, 默认不获取
}
```

响应:

```
{
    "func":"split_local_image",
    "iden":"56fc",
    "ret":0,
```

```
        "err_info":"",
        "image_path_list":["C:\\2.jpg", "C:\\3.jpg"],
        "image_base64_list:["xxx", "xxx"]" // base64 字符串，带前缀
    }
```

11. 本地生成压缩文件

请求:

```
{
    "func":"local_make_zip_file",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "file_path_list":["C:\\1.jpg", "C:\\2.jpg"],
    "local_save":true,   // 是否保存到本地，默认保存
    "get_base64":false  // 是否获取 base64，默认不获取
    "zip_path":"C:\\1.zip" // 保存文件名，默认为空，表示生成在全局目
录下
}
```

响应:

```
{
    "func":"local_make_zip_file",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "zip_path":"C:\\1.zip",
    "zip_base64":"xxx"
}
```

12. 本地图像纠偏

请求:

```
{
    "func":"local_image_deskew",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_path":"C:\\1.jpg",
    "local_save":true,   // 是否保存到本地，默认保存
    "get_base64":false  // 是否获取 base64，默认不获取
}
```

响应:

```
{
    "func":"local_image_deskew",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\2.jpg",
}
```

```
        "image_base64":"xxx"    // base64 字符串， 带前缀
    }
```

13. 本地图像添加水印

请求:

```
{
    "func":"local_image_add_watermark",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_path":"C:\\1.jpg",
    "text":"123456",
    "text_color":"#000000", // RGB
    "text_opacity":80,    // 不透明度
    "text_pos":"left",     // 可以是 left、right、top、bottom、left_top、
    right_top、left_bottom、right_bottom、center 和 location
    "margin_left":10,      // 左边距
    "margin_top":10,       // 上边距
    "margin_right":10,     // 右边距
    "margin_bottom":10,    // 下边距
    "location_x":100,      // x 坐标
    "location_y":100,      // y 坐标
    "font_name":"xxx",      // 字体名
    "font_size":20,         // 字号
    "font_bold":false,       // 粗体
    "font_underline":false,   // 下划线
    "font_italic":false,      // 斜体
    "font_strikeout":false,   // 删除线
    "local_save":true,       // 是否保存到本地， 默认保存
    "get_base64":false      // 是否获取 base64， 默认不获取
}
```

响应:

```
{
    "func":"local_image_add_watermark",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\2.jpg",
    "image_base64":"xxx"    // base64 字符串， 带前缀
}
```

14. 本地图像去污

请求:

```
{
```

```
    "func":"local_image_decontamination",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_path":"C:\\1.jpg",
    "mode":"inside",     // 可以是 inside 和 outside
    "color":"#000000",   // RGB
    "x":0,              // x
    "y":0,              // y
    "width":100,         // 宽
    "height":100,        // 高
    "local_save":true,   // 是否保存到本地，默认保存
    "get_base64":false  // 是否获取 base64，默认不获取
}
响应:
{
    "func":"local_image_add_watermark",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\2.jpg",
    "image_base64":"xxx"  // base64 字符串，带前缀
}
```

15. 本地图像方向校正

请求:

```
{
    "func":"local_image_direction_correct",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_path":"C:\\1.jpg",
    "local_save":true,   // 是否保存到本地，默认保存
    "get_base64":false  // 是否获取 base64，默认不获取
}
响应:
{
    "func":"local_image_direction_correct",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\2.jpg",
    "image_base64":"xxx"  // base64 字符串，带前缀
}
```

16. 本地图像裁剪

请求:

```
{  
    "func":"local_image_clip",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_path":"C:\\1.jpg",  
    "x":0,              // x  
    "y":0,              // y  
    "width":100,         // 宽  
    "height":100,        // 高  
    "local_save":true,   // 是否保存到本地，默认保存  
    "get_base64":false   // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func":"local_image_clip",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串，带前缀  
}
```

17. 本地图像去底色

请求:

```
{  
    "func":"local_image_fade_bkcolor",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_path":"C:\\1.jpg",  
    "local_save":true,   // 是否保存到本地，默认保存  
    "get_base64":false   // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func":"local_image_fade_bkcolor",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串，带前缀  
}
```

18. 本地图像调整色彩

请求:

```
{  
    "func":"local_image_adjust_colors",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_path":"C:\\1.jpg",  
    "brightness":0,     // 亮度, 取值-255 值 255 之间  
    "contrast":0,       // 饱和度, 取值-127 至 127 之间  
    "gamma":1.0,        // 伽马, 取值 0.1 值 5.0 之间  
    "local_save":true,   // 是否保存到本地, 默认保存  
    "get_base64":false  // 是否获取 base64, 默认不获取  
}
```

响应:

```
{  
    "func":"local_image_adjust_colors",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串, 带前缀  
}
```

19. 本地图像二值化

请求:

```
{  
    "func":"local_image_binarization",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_path":"C:\\1.jpg",  
    "local_save":true,   // 是否保存到本地, 默认保存  
    "get_base64":false  // 是否获取 base64, 默认不获取  
}
```

响应:

```
{  
    "func":"local_image_binarization",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串, 带前缀  
}
```

三、图像采集接口

1. 设备初始化

请求:

```
{  
    "func":"init_device",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}
```

响应:

```
{  
    "func":"init_device",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

事件:

```
{  
    "func":"device_arrive",  
    "iden":"56fc",  
    "device_name":"huago_scanner"  
}
```

事件:

```
{  
    "func":"device_remove",  
    "iden":"56fc",  
    "device_name":"huago_scanner"  
}
```

2. 设备反初始化（需初始化后调用）

请求:

```
{  
    "func":"deinit_device",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}
```

响应:

```
{  
    "func":"deinit_device",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

3. 获取设备是否已初始化

请求:

```
{  
    "func":"is_device_init",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}  
响应:  
{  
    "func":"is_device_init",  
    "iden":"56fc",  
    "ret":0, // 0 表示已初始化, 非 0 表示未初始化  
    "err_info":""  
}
```

4. 获取设备列表（需初始化后调用）

请求:

```
{  
    "func":"get_device_name_list",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}  
响应:  
{  
    "func":"get_device_name_list",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
    "device_name_list":["huago_scanne_1", "huago_scanne_2"]  
}
```

5. 打开设备（需初始化后调用，每次只能打开一台设备）

请求:

```
{  
    "func":"open_device",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "device_name":"huago_scanner", 为空表示第一台设备  
}  
响应:  
{  
    "func":"open_device",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

6. 关闭设备（需打开设备后调用）

请求:

```
{  
    "func":"close_device",  
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略  
}
```

响应:

```
{  
    "func":"close_device",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

7. 获取设备序列号（需打开设备后调用）

请求:

```
{  
    "func":"get_device_sn",  
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略  
}
```

响应:

```
{  
    "func":"get_device_sn",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "sn":"12345678"  
}
```

8. 获取设备固件版本号（需打开设备后调用）

请求:

```
{  
    "func":"get_device_fwversion",  
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略  
}
```

响应:

```
{  
    "func":"get_device_fwversion",  
    "iden":"56fc",  
    "ret":0,
```

```
    "err_info":"",
    "fwversion":"12345678"
}
```

9. 设置设备参数（需打开设备后和未开始扫描时调用，设置的时候有些参数可能成功，有些参数可能失败，有一个参数设置失败时就会返回非 0，并设置相关的错误信息）

请求:

```
{
    "func":"set_device_param",
    "iden":"56fc",          // 可以是随机字符串，该字段可以省略
    "device_param":
    [
        {
            "name":"xxx",
            "value":"xxx"
        },
        {
            "name":"xxx",
            "value":0
        },
        {
            "name":"xxx",
            "value":0.0000
        },
        {
            "name":"xxx",
            "value":true
        }
    ]
}
```

响应:

```
{
    "func":"set_device_param",
    "iden":"56fc",
    "ret":0,
    "err_info":""
}
```

10. 获取设备参数（需打开设备后调用）

请求:

```
{
```

```
    "func":"get_device_param",
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略
}
响应：
{
    "func":"get_device_param",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "device_param":
    [
        {
            "group_name":"xxx",
            "group_param":
            [
                {
                    "name":"xxx",
                    "value_type":"string",
                    "value":"xxx",
                    "range_type":"list",
                    "value_list":["xxx", "xxx"]
                },
                {
                    "name":"xxx",
                    "value_type":"int",
                    "value":100,
                    "range_type":"list",
                    "value_list":[100, 200]
                },
                {
                    "name":"xxx",
                    "value_type":"int",
                    "value":20,
                    "range_type":"min_max",
                    "value_min":10,
                    "value_max":30
                },
                {
                    "name":"xxx",
                    "value_type":"bool",
                    "value":true
                }
            ]
        }
    ]
}
```

```
        },
        {
            // 其他组
        }
    ]
}
```

11. 重置设备参数（需打开设备后和未开始扫描时调用）

请求:

```
{
    "func":"reset_device_param",
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略
}
```

响应:

```
{
    "func":"reset_device_param",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

12. 获取当前设备名称（需打开设备后调用）

请求:

```
{
    "func":"get_curr_device_name",
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略
}
```

响应:

```
{
    "func":"get_curr_device_name",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "device_name":"huago_scanner"
}
```

13. 开始扫描（需打开设备后调用）

请求:

```
{
    "func":"start_scan",
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略
}
```

```

    "blank_check": false, // 是否进行空白页检测, 默认不检测
    "local_save":true,   // 是否保存到本地, 默认保存
    "get_base64":false // 是否获取 base64, 默认不获取
}
响应:
{
    "func":"start_scan",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
事件:
{
    "func":"scan_begin",
    "iden":"56fc"
}
事件:
{
    "func":"scan_end",
    "iden":"56fc"
}
事件:
{
    "func":"scan_info",
    "iden":"56fc",
    "is_error":false,           // 是否是错误消息
    "info":"start scanning"   // 消息内容
}
事件:
{
    "func":"scan_image",
    "iden":"56fc",
    "is_blank":false, // 是否是空白页
    "image_path":"C:\\1.jpg", // 扫描图片的本地路径
    "image_base64":"xxx"   // base64 字符串, 带前缀
}

```

14. 停止扫描（需开始扫描后调用）

请求:

```

{
    "func":"stop_scan",
    "iden":"56fc"        // 可以是随机字符串, 该字段可以省略
}

```

响应:

```
{  
    "func":"stop_scan",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

15. 获取设备是否正在扫描

请求:

```
{  
    "func":"is_device_scanning",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}
```

响应:

```
{  
    "func":"is_device_scanning",  
    "iden":"56fc",  
    "ret":0, // 0 表示正在扫描, 非 0 表示未进行扫描  
    "err_info":""  
}
```

四、图像管理接口

1. 获取批次号列表

请求:

```
{  
    "func":"get_batch_id_list",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}
```

响应:

```
{  
    "func":"get_batch_id_list",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
    "batch_id_list":["default","cbf1","efc2"]  
}
```

2. 打开批次 (后端启动后默认打开最近批次, 没有最近批次则会新建 default 批次并打开)

请求:

```
{  
    "func":"open_batch",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "batch_id":"default" // 批次 id  
}
```

响应:

```
{  
    "func":"open_batch",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

3. 删除批次

请求:

```
{  
    "func":"delete_batch",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "batch_id":"default" // 批次 id, 不能是当前批次  
}
```

响应:

```
{  
    "func":"delete_batch",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

4. 新建批次（新建批次到列表尾部并打开该批次）

请求:

```
{  
    "func":"new_batch",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "batch_id":"001"    // 批次 id, 如果为空则会自动生成随机值  
}
```

响应:

```
{  
    "func":"new_batch",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

```
    "err_info":"",
    "batch_id":"001"
}
```

5. 获取当前批次号

请求:

```
{
    "func":"get_curr_batch_id",
    "iden":"56fc"      // 可以是随机字符串，该字段可以省略
}
```

响应:

```
{
    "func":"get_curr_batch_id",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "batch_id":"001"    // 批次 id
}
```

6. 修改批次号

请求:

```
{
    "func":"modify_batch_id",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "batch_id":"001",   // 批次 id
    "new_batch_id":"002" // 新的 id
}
```

响应:

```
{
    "func":"modify_batch_id",
    "iden":"56fc",
    "ret":0,
    "err_info":""
}
```

7. 绑定文件夹

请求:

```
{
    "func":"bind_folder",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "folder":"C:\\\"     // 文件夹
}
```

```
    "name_mode":"order", // 命名方式, 目前仅支持 order
    "name_width":1,     // 名字宽度, 默认为 1
    "name_base":0       // 默认为 0
}
响应:
{
    "func":"bind_folder",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

8. 停止绑定文件夹

请求:

```
{
    "func":"stop_bind_folder",
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略
}
响应:
{
    "func":"stop_bind_folder",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

9. 加载图像缩略图列表

请求:

```
{
    "func":"get_image_thumbnail_list",
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略
}
响应:
{
    "func":"get_image_thumbnail_list",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_thumbnail_list":[{"image_tag":"001", "image_base64":"xxx"}]
}
```

10. 获取图像数量

请求:

```
{  
    "func":"get_image_count",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}
```

响应:

```
{  
    "func":"get_image_count",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    "image_count":2    // 图像数量  
}
```

11. 加载图像

请求:

```
{  
    "func":"load_image",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_index":0     // 图像索引, 从 0 开始  
}  
响应:  
{  
    "func":"load_image",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    "image_tag":"001",  
    "image_base64":"xxx"  
}
```

12. 保存图像

请求:

```
{  
    "func":"save_image",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_index":0     // 图像索引, 从 0 开始  
}  
响应:  
{
```

```
    "func":"save_image",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\1.jpg"
}
```

13. 插入本地图像（需返回成功后前端缩略图才能作相应的 UI 修改）

请求:

```
{
    "func":"insert_local_image",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_path":"C:\\1.jpg",
    "insert_pos": -1,   // 插入位置， -1 表示最后
    "image_tag":"001"   // 标签名，可以为空
}
```

响应:

```
{
    "func":"insert_local_image",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
}
```

14. 插入图像（需返回成功后前端缩略图才能作相应的 UI 修改）

请求:

```
{
    "func":"insert_image",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_base64":"xxx", // 图像 base64，带前缀
    "insert_pos": -1,   // 插入位置， -1 表示最后
    "image_tag":"001"   // 标签名，可以为空
}
```

响应:

```
{
    "func":"insert_image",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
}
```

15. 修改图像标签（需返回成功后前端缩略图才能作相应的 UI 修改）

请求:

```
{  
    "func":"modify_image_tag",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_index_list":[0,1,2],  
    "image_tag_list":["001","002","003"]  
}
```

响应:

```
{  
    "func":"modify_image_tag",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

16. 删除图像 (需返回成功后前端缩略图才能作相应的 UI 修改)

请求:

```
{  
    "func":"delete_image",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_index_list":[0,1,2]  
}
```

响应:

```
{  
    "func":"delete_image",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

17. 清理图像列表 (需返回成功后前端缩略图才能作相应的 UI 修改)

请求:

```
{  
    "func":"clear_image_list",  
    "iden":"56fc"      // 可以是随机字符串, 该字段可以省略  
}
```

响应:

```
{  
    "func":"clear_image_list",  
    "iden":"56fc",  
    "ret":0,  
}
```

```
        "err_info":""  
    }
```

18. 修改图像（需返回成功后前端缩略图才能作相应的 UI 修改）

请求:

```
{  
    "func":"modify_image",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_index":0,  
    "image_base64":"xxx"  
}
```

响应:

```
{  
    "func":"modify_image",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

19. 使用本地图像修改图像（需返回成功后前端缩略图才能作相应的 UI 修改）

请求:

```
{  
    "func":"modify_image_by_local",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_index":0,  
    "image_path":"C:\\1.jpg"  
}
```

响应:

```
{  
    "func":"modify_image_by_local",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":""  
}
```

20. 移动图像（需返回成功后前端缩略图才能作相应的 UI 修改，如果前端不知道如何修改 UI，可以刷新图像列表）

请求:

```
{  
    "func":"move_image",
```

```
        "iden":"56fc",      // 可以是随机字符串，该字段可以省略
        "image_index_list":[0,1,2],
        "mode":"pos",       // pos 或 index, 默认为 index
        "target":0
    }
响应:
{
    "func":"move_image",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

21. 交换图像（需返回成功后前端缩略图才能作相应的 UI 修改）

请求:

```
{
    "func":"exchange_image",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_index_1":0,
    "image_index_2":1
}
响应:
{
    "func":"exchange_image",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

22. 图像书籍排序（需返回成功后前端缩略图才能作相应的 UI 修改，如果前端不知道如何修改 UI，可以刷新图像列表）

请求:

```
{
    "func":"image_book_sort",
    "iden":"56fc"        // 可以是随机字符串，该字段可以省略
}
响应:
{
    "func":"image_book_sort",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

```
}
```

23. 上传图像

请求:

```
{
    "func":"upload_image",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_index":0,
    "remote_file_path":"/path/1.jpg",    // 远程文件路径
    "upload_mode":"http",  // http 或 ftp
    "http_host":"192.168.1.100",
    "http_port":80,
    "http_path":"/upload.php",
    "ftp_user":"xxx",
    "ftp_password":"xxx",
    "ftp_host":"192.168.1.100",
    "ftp_port":21
}
```

响应:

```
{
    "func":"upload_image",
    "iden":"56fc",
    "ret":0,
    "err_info": ""
}
```

24. 合成图像

请求:

```
{
    "func":"merge_image",
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略
    "image_index_list":[0,1,2],
    "mode":"horz",     // 可以是 horz 和 vert
    "align":"top",      // 可以是 top, bottom, center, left 和 right
    "interval":0,       // 间隔
    "local_save":true,  // 是否保存到本地，默认保存
    "get_base64":false // 是否获取 base64，默认不获取
}
```

响应:

```
{
    "func":"merge_image",
    "iden":"56fc",
```

```
    "ret":0,  
    "err_info": "",  
    "image_path": "C:\\1.jpg",  
    "image_base64": "xxx" // base64 字符串，带前缀  
}
```

25. 合成多张图像

请求:

```
{  
    "func": "make_multi_image",  
    "iden": "56fc", // 可以是随机字符串，该字段可以省略  
    "image_index_list": [0,1,2],  
    "format": "tif", // 格式，支持 tif、pdf 和 odf  
    "tiff_compression": "none", // 可以为 none、lzw 和 jpeg  
    "tiff_jpeg_quality": 80, // tiff-jpeg 质量  
    "local_save": true, // 是否保存到本地，默认保存  
    "get_base64": false // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func": "make_multi_image",  
    "iden": "56fc",  
    "ret": 0,  
    "err_info": "",  
    "image_path": "C:\\1.jpg",  
    "image_base64": "xxx" // base64 字符串，带前缀  
}
```

26. 拆分图像

请求:

```
{  
    "func": "split_image",  
    "iden": "56fc", // 可以是随机字符串，该字段可以省略  
    "image_index": 0,  
    "mode": "horz", // horz,vert  
    "location": 1000,  
    "local_save": true, // 是否保存到本地，默认保存  
    "get_base64": false // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func": "split_image",
```

```
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path_list":["C:\\1.jpg", "C:\\2.jpg"],
    "image_base64_list":"xxx" // base64 字符串，带前缀
}
```

27. 压缩图像

请求:

```
{
    "func":"make_zip_file",
    "iden":"56fc", // 可以是随机字符串，该字段可以省略
    "image_index_list":[0,1,2],
    "local_save":true, // 是否保存到本地，默认保存
    "get_base64":false // 是否获取 base64，默认不获取
}
```

响应:

```
{
    "func":"make_zip_file",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "zip_path":"C:\\1.zip",
    "zip_base64":"xxx"
}
```

28. 图像纠偏

请求:

```
{
    "func":"image_deskew",
    "iden":"56fc", // 可以是随机字符串，该字段可以省略
    "image_index":0,
    "local_save":true, // 是否保存到本地，默认保存
    "get_base64":false // 是否获取 base64，默认不获取
}
```

响应:

```
{
    "func":"image_deskew",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\1.jpg",
```

```
        "image_base64":"xxx"    // base64 字符串， 带前缀  
    }
```

29. 图像添加水印

请求:

```
{  
    "func":"image_add_watermark",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_index":0,  
    "text":"123456",  
    "text_color":"#000000", // RGB  
    "text_opacity":80,    // 不透明度  
    "text_pos":"left",     // 可以是 left、right、top、bottom、left_top、  
    right_top、left_bottom、right_bottom、center 和 location  
    "margin_left":10,      // 左边距  
    "margin_top":10,      // 上边距  
    "margin_right":10,     // 右边距  
    "margin_bottom":10,    // 下边距  
    "location_x":100,      // x 坐标  
    "location_y":100,      // y 坐标  
    "font_name":"xxx",      // 字体名  
    "font_size":20,        // 字号  
    "font_bold":false,      // 粗体  
    "font_underline":false, // 下划线  
    "font_italic":false,    // 斜体  
    "font_strikeout":false, // 删除线  
    "local_save":true,      // 是否保存到本地， 默认保存  
    "get_base64":false     // 是否获取 base64, 默认不获取  
}
```

响应:

```
{  
    "func":"image_add_watermark",  
    "iden":"56fc",  
    "ret":0,  
    "err_info":'',  
    "image_path":"C:\\1.jpg",  
    "image_base64":"xxx"    // base64 字符串， 带前缀  
}
```

30. 图像去污

请求:

```
{
```

```
    "func":"image_decontamination",
    "iden":"56fc",          // 可以是随机字符串，该字段可以省略
    "image_index":0,
    "mode":"inside",        // 可以是 inside 和 outside
    "color":"#000000",       // RGB
    "x":0,                  // x
    "y":0,                  // y
    "width":100,             // 宽
    "height":100,            // 高
    "local_save":true,      // 是否保存到本地，默认保存
    "get_base64":false     // 是否获取 base64，默认不获取
}
响应:
{
    "func":"image_decontamination",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\2.jpg",
    "image_base64":"xxx"   // base64 字符串，带前缀
}
```

31. 图像方向校正

请求:

```
{
    "func":"image_direction_correct",
    "iden":"56fc",          // 可以是随机字符串，该字段可以省略
    "image_index":0,
    "local_save":true,      // 是否保存到本地，默认保存
    "get_base64":false     // 是否获取 base64，默认不获取
}
响应:
{
    "func":"image_direction_correct",
    "iden":"56fc",
    "ret":0,
    "err_info":"",
    "image_path":"C:\\1.jpg",
    "image_base64":"xxx"   // base64 字符串，带前缀
}
```

32. 图像裁剪

请求:

```
{  
    "func":"image_clip",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_index":0,  
    "x":0,              // x  
    "y":0,              // y  
    "width":100,         // 宽  
    "height":100,        // 高  
    "local_save":true,   // 是否保存到本地，默认保存  
    "get_base64":false   // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func":"image_clip",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串，带前缀  
}
```

33. 图像去底色

请求:

```
{  
    "func":"image_fade_bkcolor",  
    "iden":"56fc",      // 可以是随机字符串，该字段可以省略  
    "image_index":0,  
    "local_save":true,   // 是否保存到本地，默认保存  
    "get_base64":false   // 是否获取 base64，默认不获取  
}
```

响应:

```
{  
    "func":"image_fade_bkcolor",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串，带前缀  
}
```

34. 图像调整色彩

请求:

```
{  
    "func":"image_adjust_colors",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_index":0,  
    "brightness":0,     // 亮度, 取值-255 值 255 之间  
    "contrast":0,       // 饱和度, 取值-127 至 127 之间  
    "gamma":1.0,        // 伽马, 取值 0.1 值 5.0 之间  
    "local_save":true,  // 是否保存到本地, 默认保存  
    "get_base64":false  // 是否获取 base64, 默认不获取  
}
```

响应:

```
{  
    "func":"image_adjust_colors",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串, 带前缀  
}
```

35. 图像二值化

请求:

```
{  
    "func":"image_binarization",  
    "iden":"56fc",      // 可以是随机字符串, 该字段可以省略  
    "image_index":0,  
    "local_save":true,  // 是否保存到本地, 默认保存  
    "get_base64":false  // 是否获取 base64, 默认不获取  
}
```

响应:

```
{  
    "func":"image_binarization",  
    "iden":"56fc",  
    "ret":0,  
    "err_info": "",  
    "image_path":"C:\\2.jpg",  
    "image_base64":"xxx"  // base64 字符串, 带前缀  
}
```