

# 基于图像识别的远程自动化测试方案

QA中心-测试开发部  
刘双

# 目录

- 传统自动化测试方案
- 图像识别引擎
- 远程工具Guacamole介绍与改进
- 用例录制与执行
- 总结展望

# 传统自动化测试—元素识别

抓取元素

- spy++ 抓取被测窗口元素信息

查找窗口

- win32gui.FindWindowEx  
查找窗口句柄hwnd

定位坐标

- win32gui.GetWindowRect(hwnd)  
获取窗口坐标

执行动作

- win32api.mouse\_event
- windll.user32.SendInput

# 传统方式存在的问题

被测软件使用自定义  
界面库，无法获取界  
面元素信息

被测软件屏蔽了鼠标  
键盘的脚本操作，导  
致用例执行失败

脚本执行受被测系统  
权限限制

开发成本较高





# 新思路

360技术嘉年华

图像识别引擎



开源的HTML5远程解  
决方案Guacamole



基于图像识别的远程自动化测试方案



# 图像识别

目标图像



当前测试端屏幕图像



图像识别引擎

坐标: 35,336

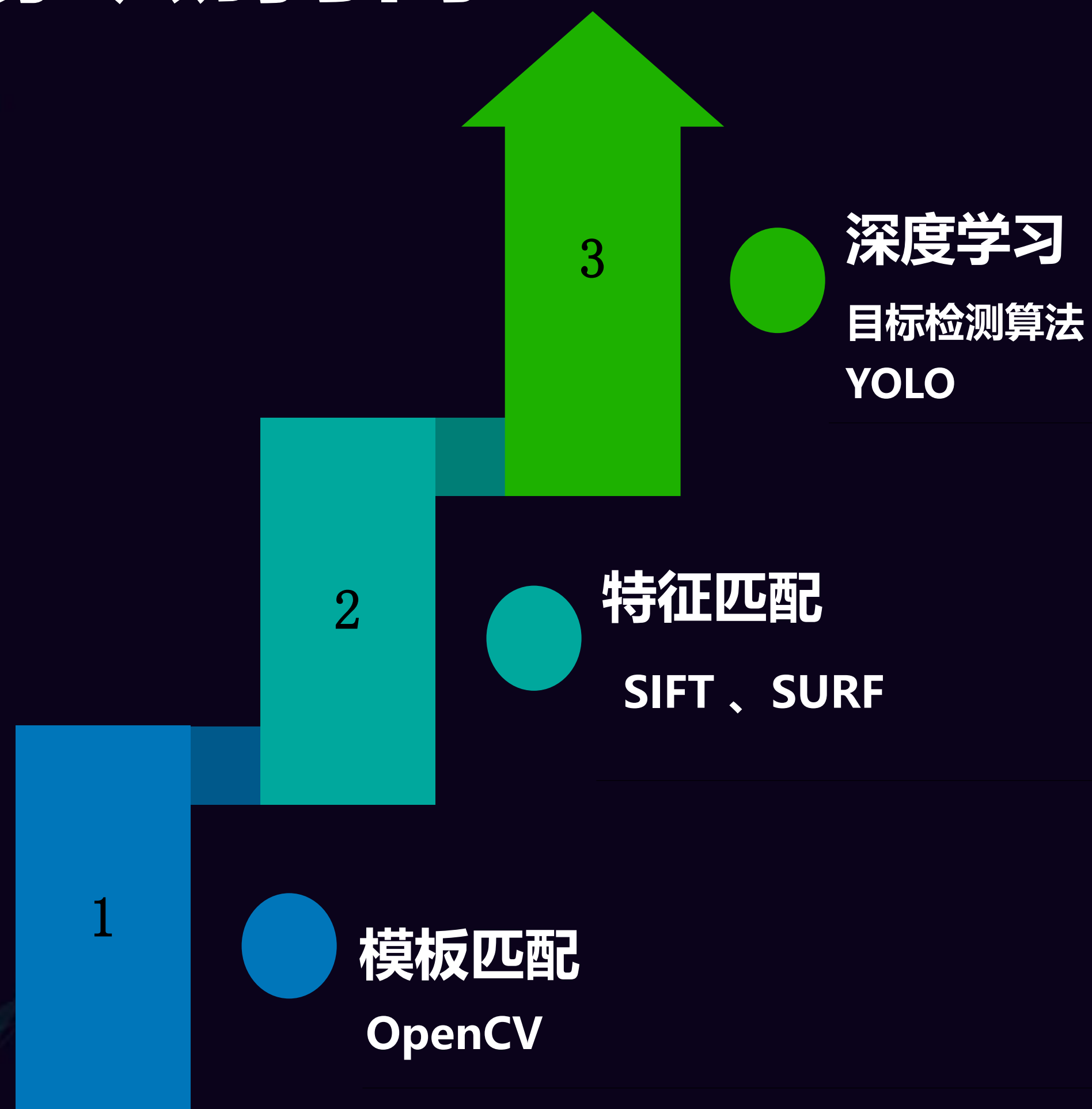
# 图像识别自动化测试的常用工具

|         | SikuliX                      | Airtest               |
|---------|------------------------------|-----------------------|
| UI定位    | 图像匹配                         | 图像匹配+UI检视             |
| 是否收费    | 免费                           | 免费                    |
| 语言      | Jython, java                 | python                |
| 平台      | Windows, Mac, Linux, Android | Windows, Android, IOS |
| 是否安装客户端 | 是                            | 是                     |
| 图像识别    | OpenCV                       | OpenCV 模板匹配+SIFT特征值匹配 |



# 图像识别引擎

360技术嘉年华



## ➤ 模板匹配

基于像素，算法相对简单，速度快

## ➤ 特征匹配

基于区域，算法相对复杂，适应于匹配目标发生旋转或大小变化的情况

## ➤ 深度学习

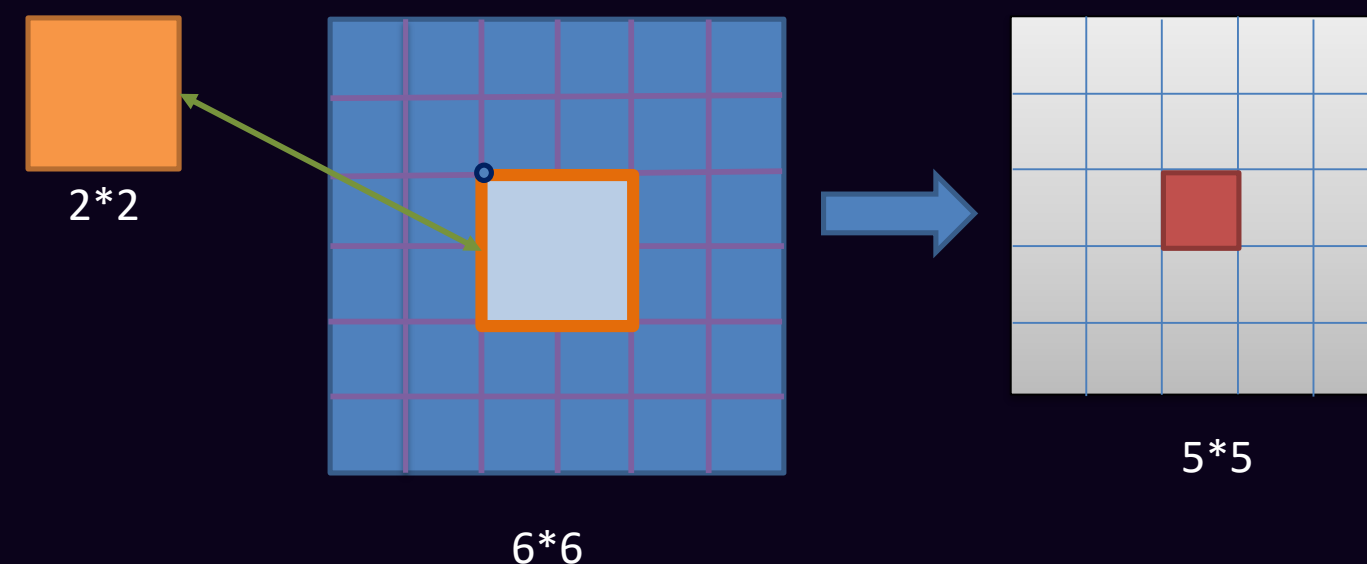
对数据要求高，训练时间较长

优先模板匹配，然后特征匹配，最后选择深度学习



# 模板匹配

- 基于像素的匹配，用模板图像在目标图像上滑动
- 对模板图像和目标图像的子区域进行比较
- 返回的结果是一个  $(W-w+1) \times (H-h+1)$  灰度图像，每一个像素值表示此区域与模板的匹配程度

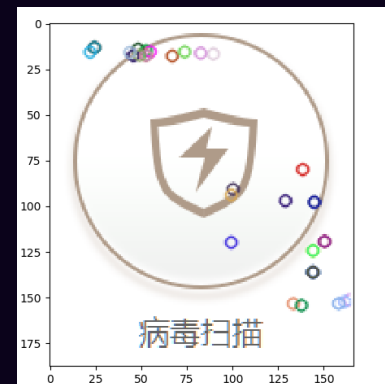


局限：只能进行平行移动，若原图像中的匹配目标发生旋转或大小变化，匹配失败

# 特征匹配

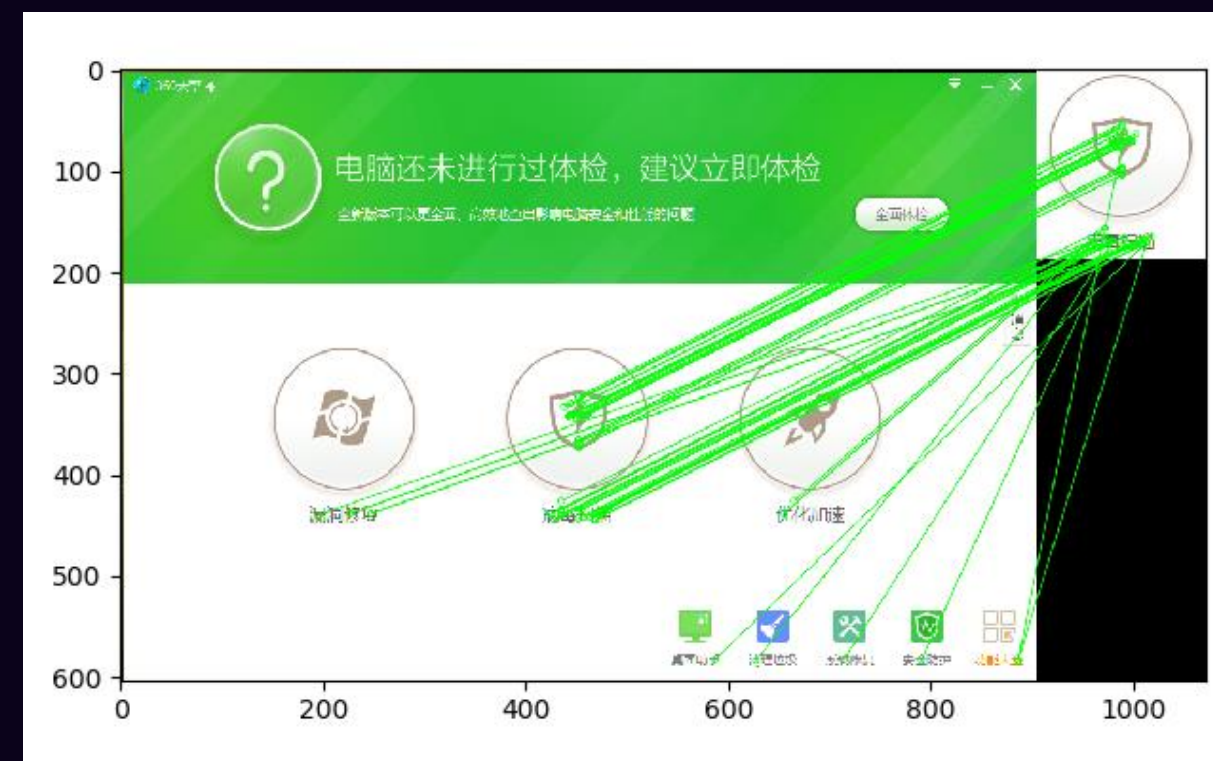
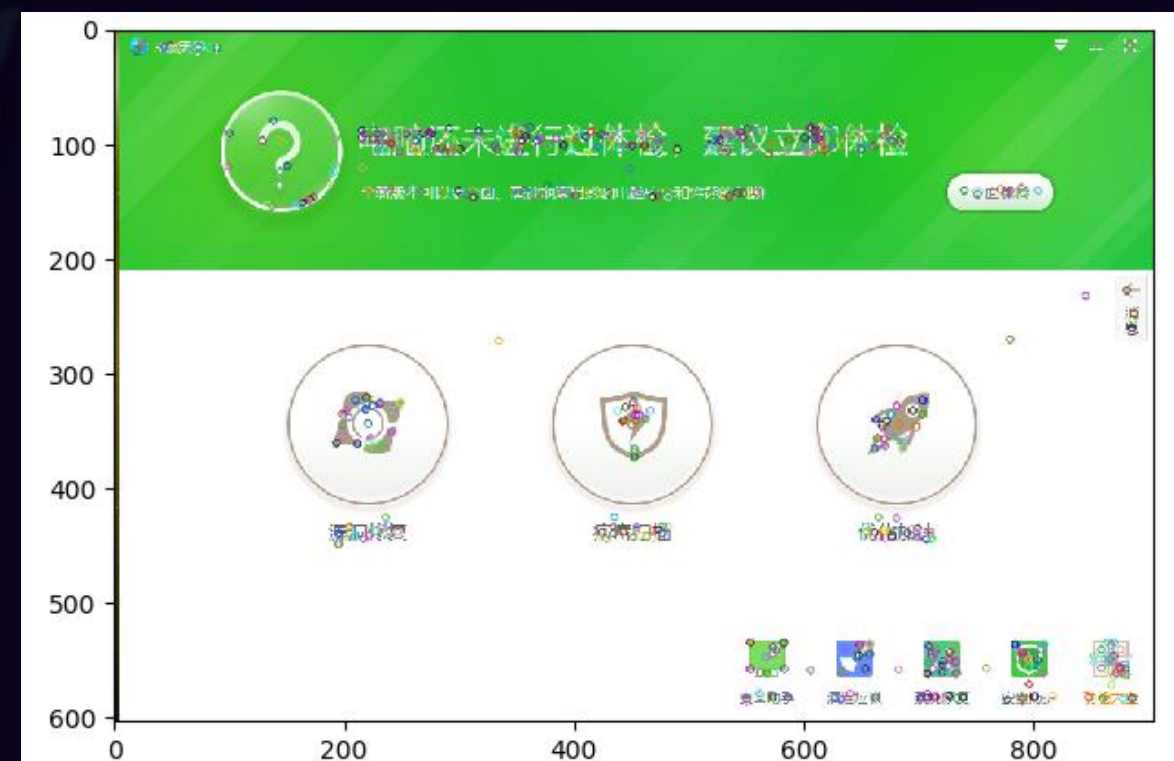
图像中提取许多特征，确保即使旋转/缩放/倾斜也能识别相同的特征

SIFT、SURF提取图像特征



FLANN特征匹配

将当前帧中匹配度最高的对象找出



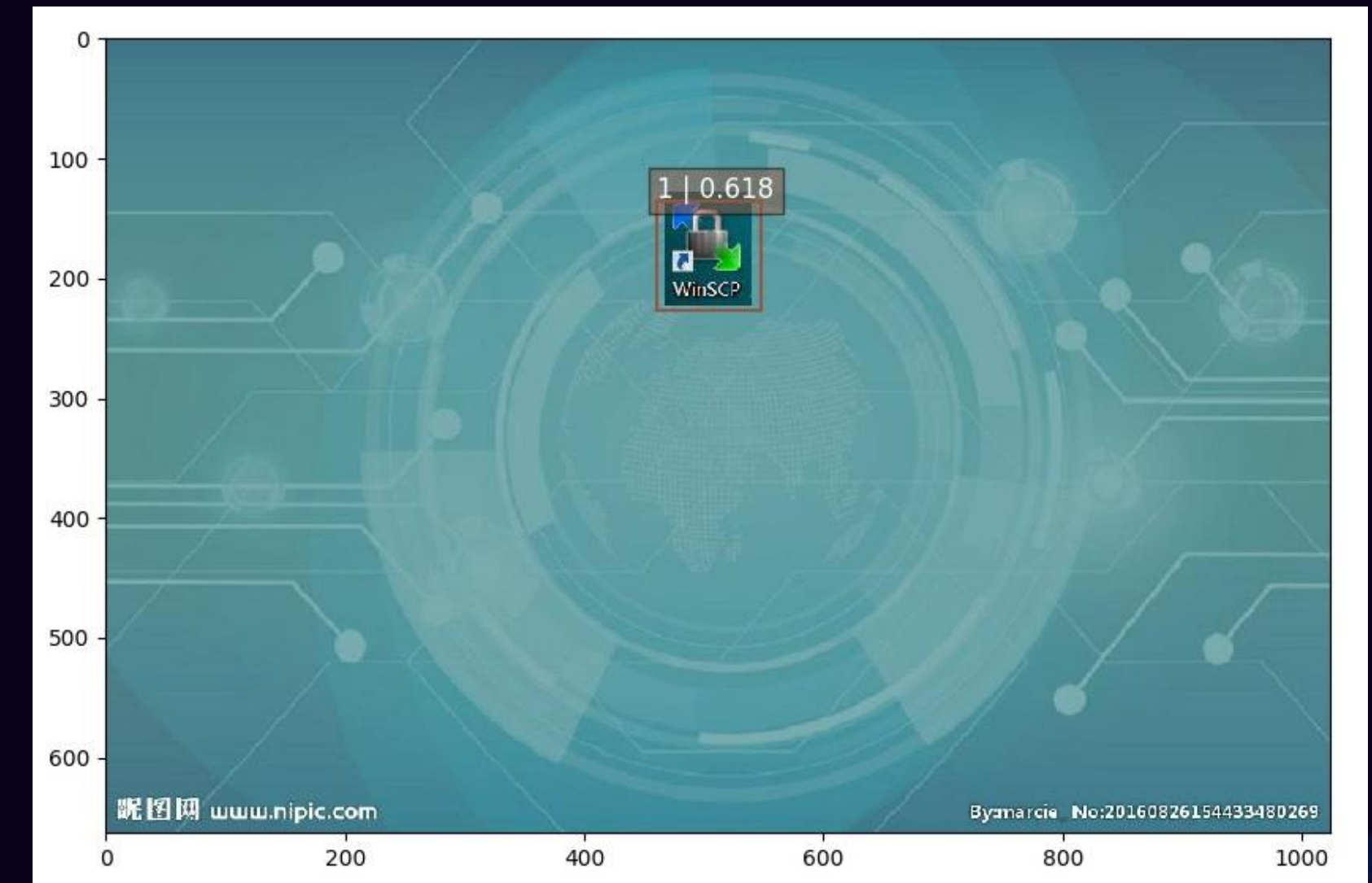
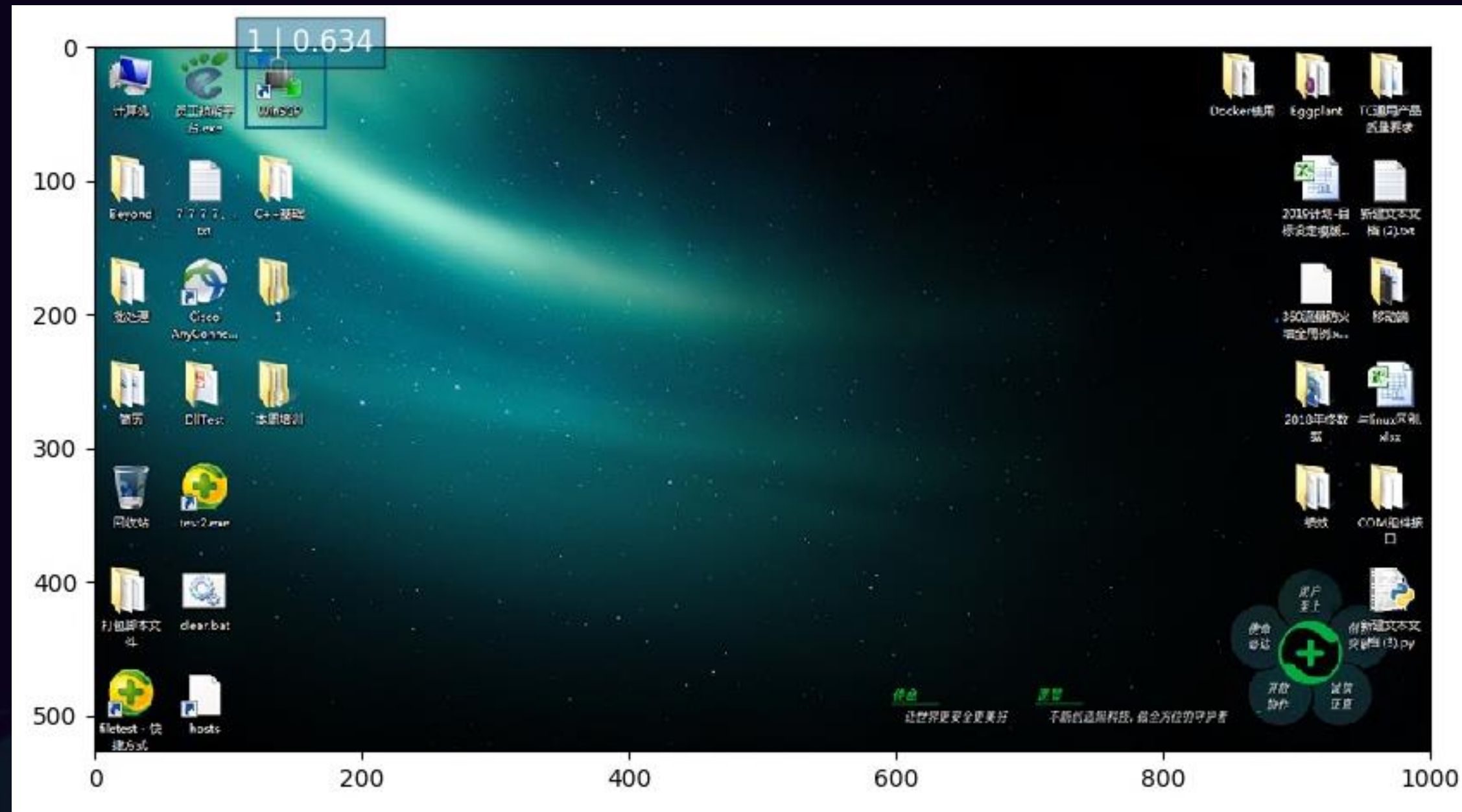
单应性变换(Homography)  
一个平面到另一个平面的映射关系



# 深度学习

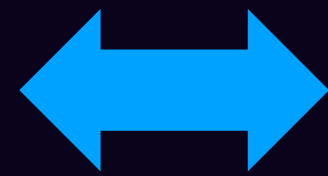
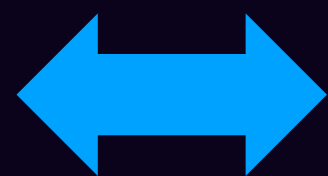
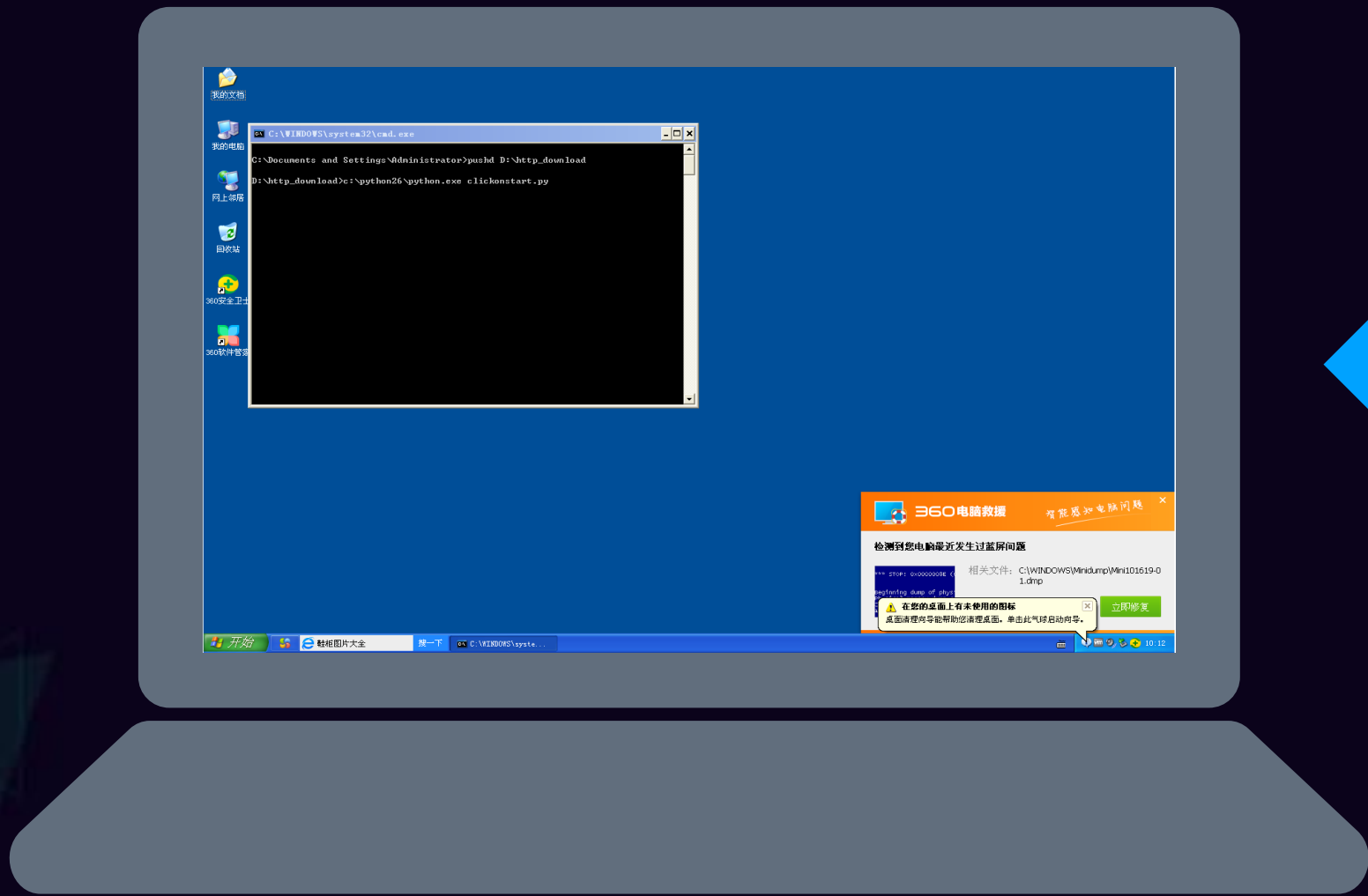
360技术嘉年华

- 准备数据：标记要训练的图像，进行分类
- 训练：用YOLO模型训练数据
- 执行测试：用训练好的模型执行测试，识别出目标图片的分类

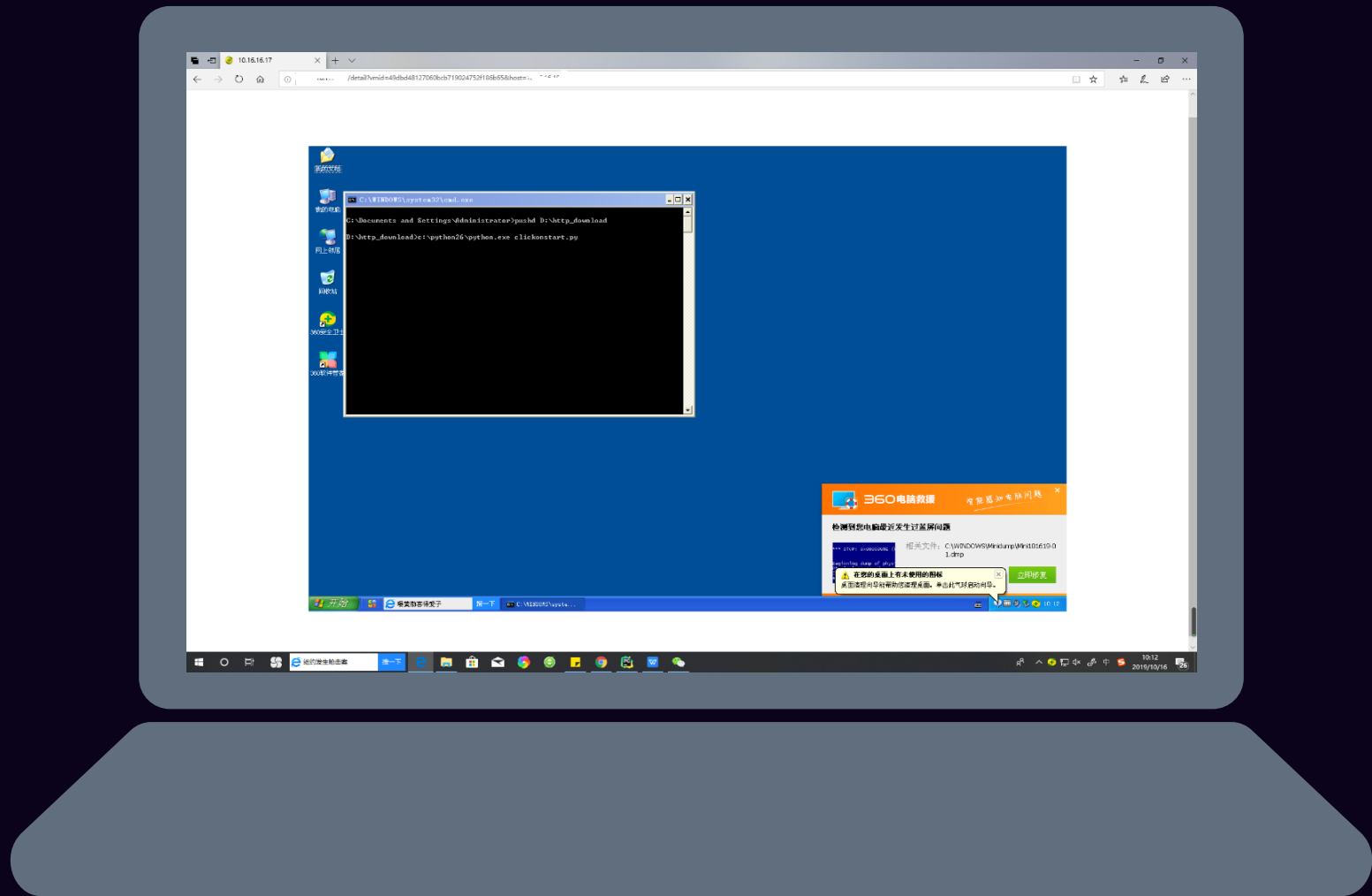


YOLO训练模型的效果

# Guacamole

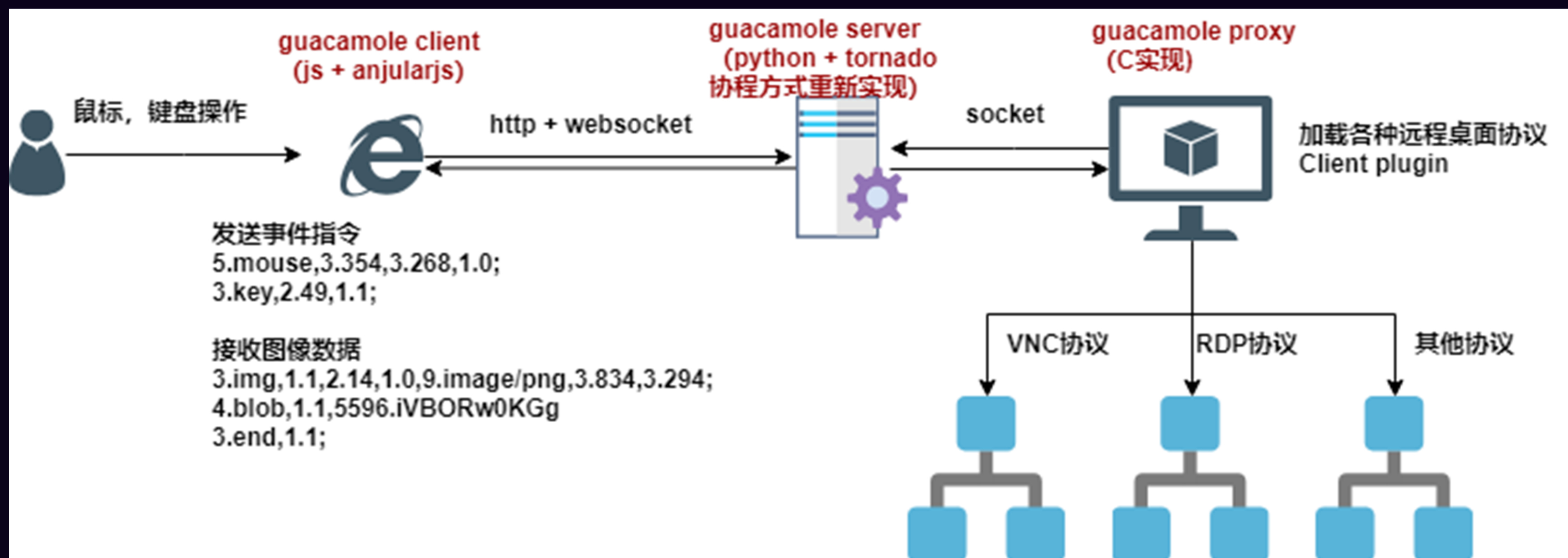


**Guacamole**  
浏览器端访问远程机器的  
工具



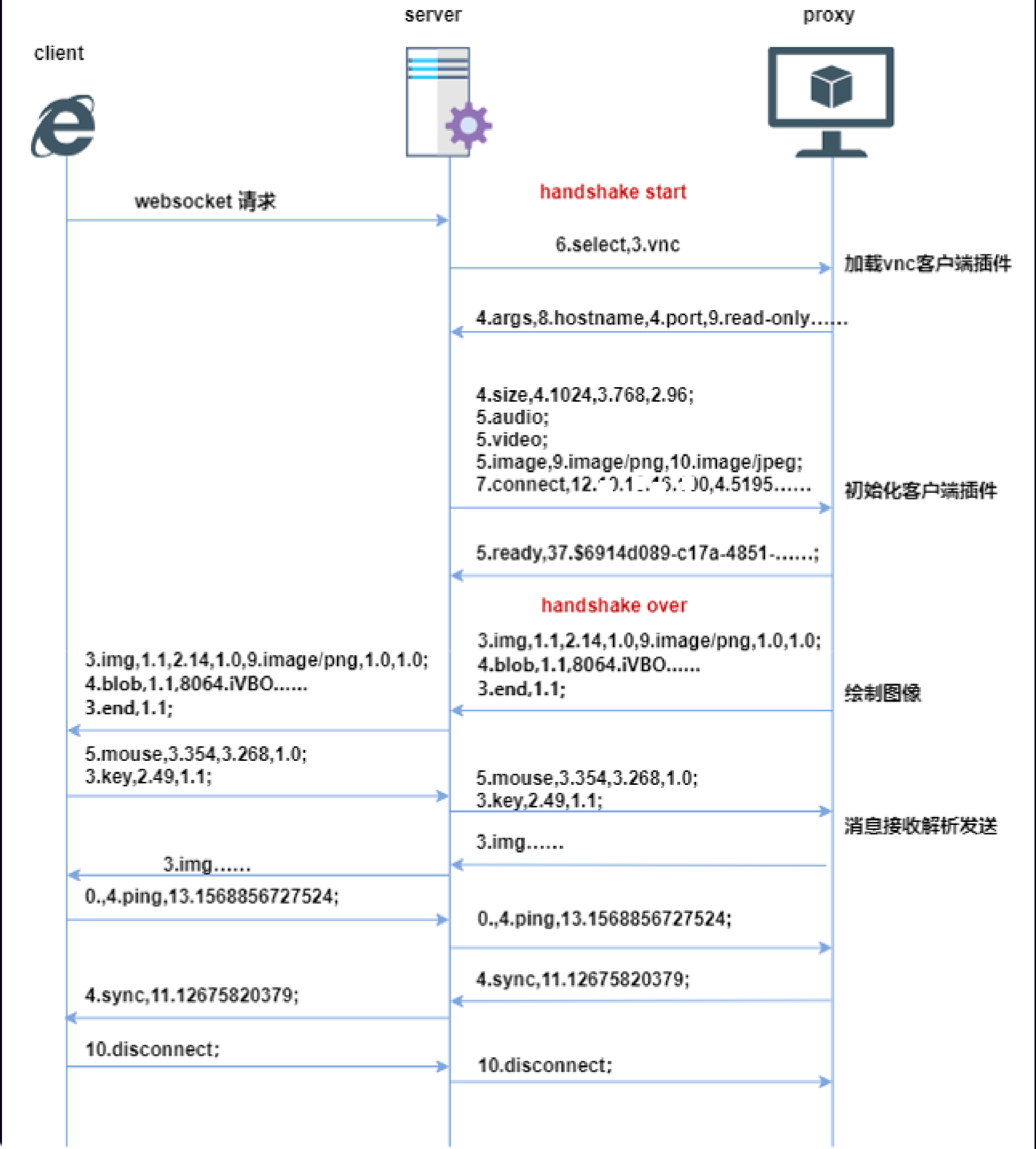


# Guacamole结构介绍



**Guacamole协议格式:**  
操作码, 参数1, 参数2, 参数3, .....;  
指令内部每个元素构成: 长度.值

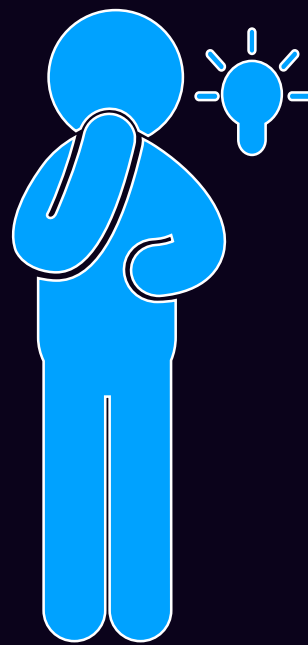
# Guacamole原理



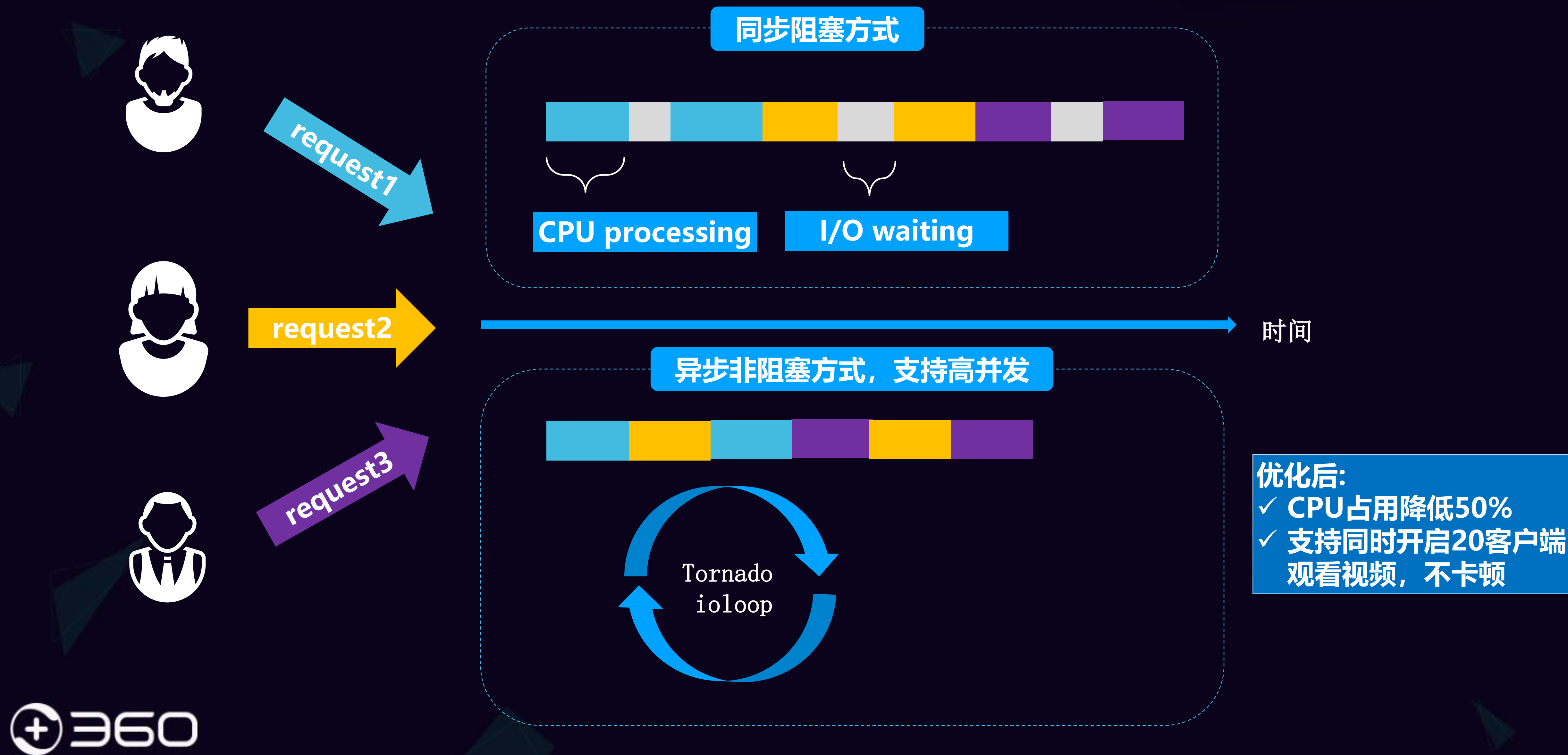
Guacamole server存在的问题:

与现有虚拟机云平台结合困难,

并发大时存在性能瓶颈



# 改造guacamole





# Guacamole功能扩展

360技术嘉年华

## ✓ 鼠标键盘操作接口化

对外提供鼠标键盘远程操作api接口，服务已有的自动化用例执行平台

## ✓ 远程桌面区域截图

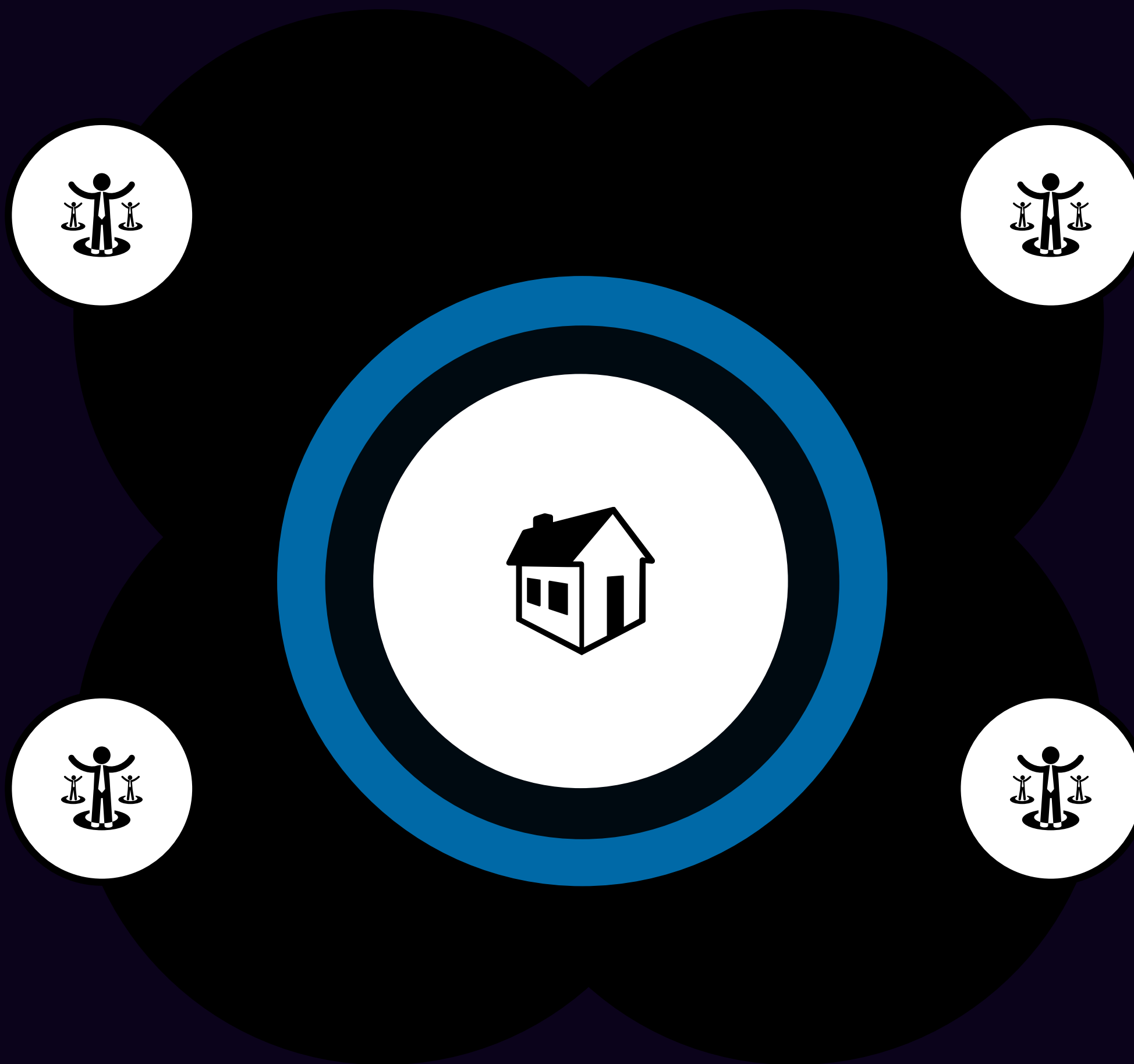
修改Guacamole javascript代码，实现对远程桌面的区域截图

## ✓ 远程机器文件上传下载

通过消息队列与虚拟机云平台通信，实现与远程虚拟机的文件交互

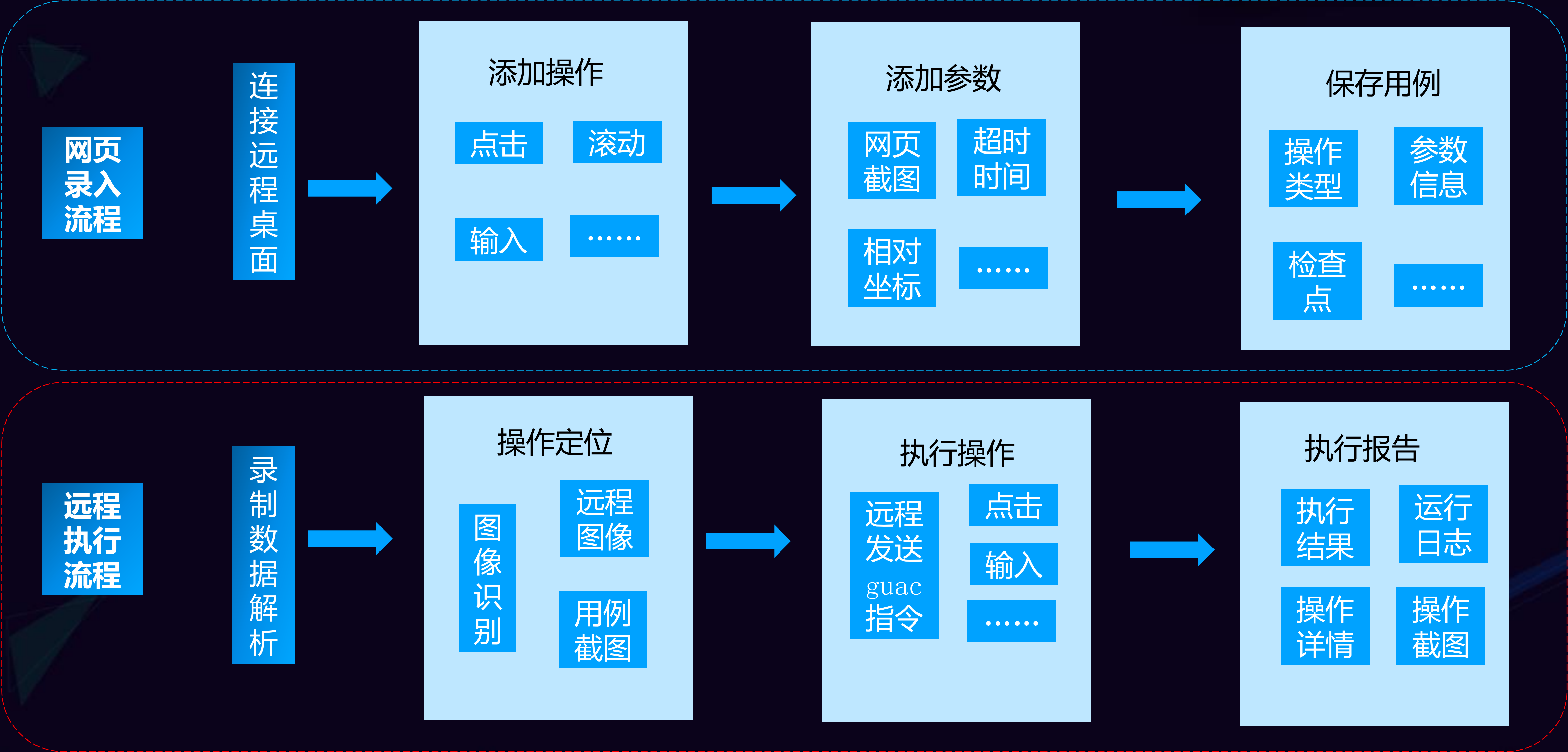
## ✓ 客户端用例录制

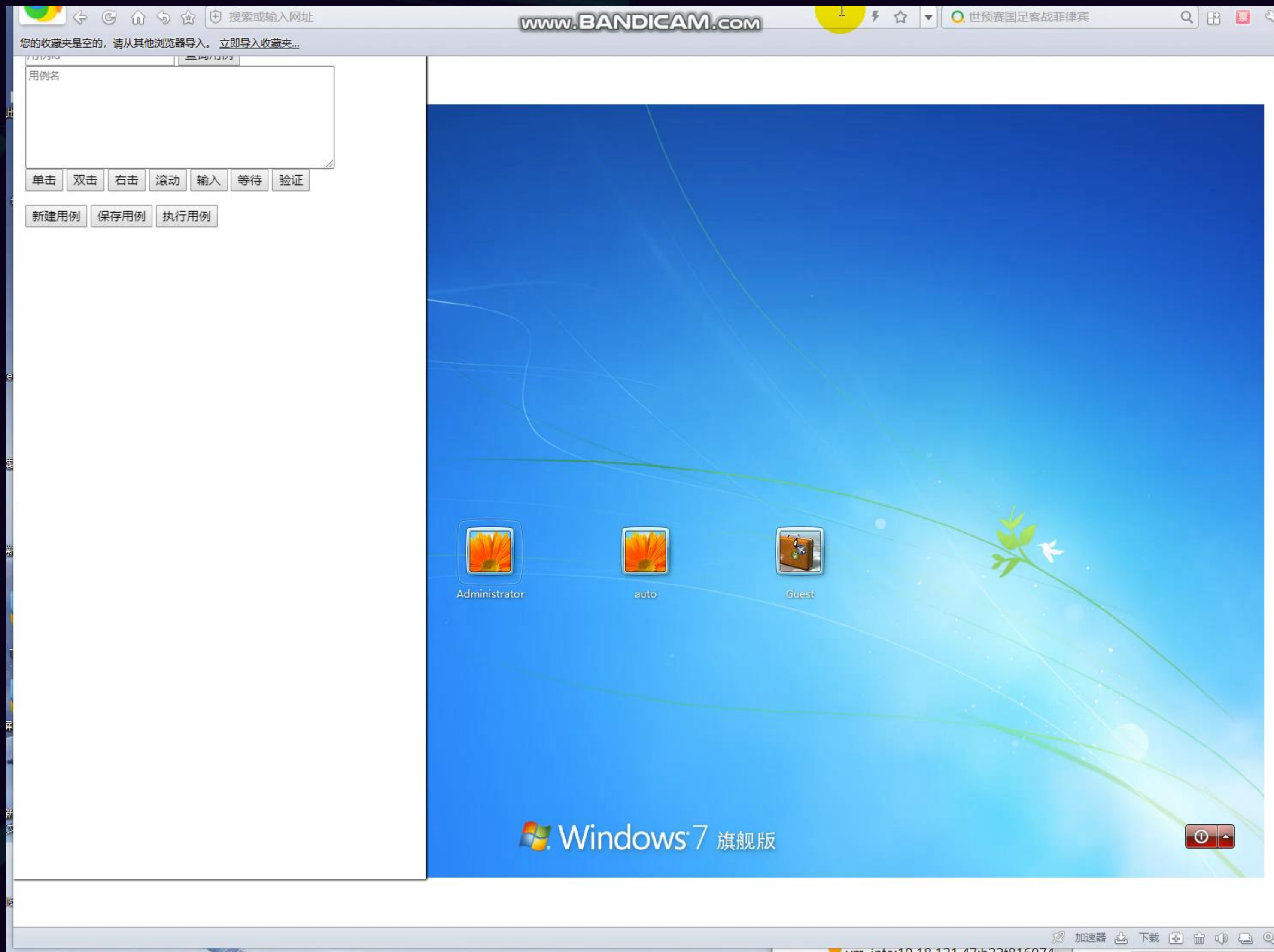
jquery + angularjs实现浏览器端用例录制和回放





# 用例录入和执行





# 传统方案 VS 新方案

测试脚本在被测机器上执行，主要基于控件识别

## 优势

- 速度快，稳定性强，更灵活

## 劣势

- 脚本容易被拦截
- 鼠标键盘操作失效
- 机器重启，脚本也会退出
- 自动化用例开发速度慢

传统方案

新方案

测试脚本在被测机器外围执行，基于图像识别

## 优势

- 脱离系统权限，不受限于UAC
- 脚本不中断的情况下，操作系统登录、重启、安全模式、蓝屏等状态
- 能快速实现自动化用例

## 劣势

- 系统稳定性依赖图像识别引擎效果
- 执行速度较传统方式慢



# 总结

360技术嘉年华

新方案不能替代传统自动化测试方案，而应该作为一种补充

需要快速开展自动化测试的新业务

界面大部分由图片构成的应用，如  
游戏应用

特殊的测试场景，如开启UAC，  
操作安全模式，判断蓝屏等

Linux，Mac等操作系统上的自动  
化测试

应用场景



# 展望

通用操作函数化，降低  
用例编写和维护成本

与现有自动化测试平台结合

提高图像识别的成功率

文字识别功能



谢谢！