

SKEye-Unity-SDK 说明文档

1. 概述

[SKEye](#) 是[盛开互动](#)依托十余年研发积淀推出的图像识别技术平台，致力于为智能产品、应用提供精准快速的“云+端”图像识别 SDK。SKEyeSDK 是一套基于 SKEye 人工智能开放平台的图像识别服务，该文档将详细说明如何在 Unity 平台上快速集成该 SDK，并通过调用 SDK 接口使用 SKEye 图像识别服务。

2. 系统需求

操作系统 Win7 Win8 Win10 MacOS 10.x Unity3d 64 位

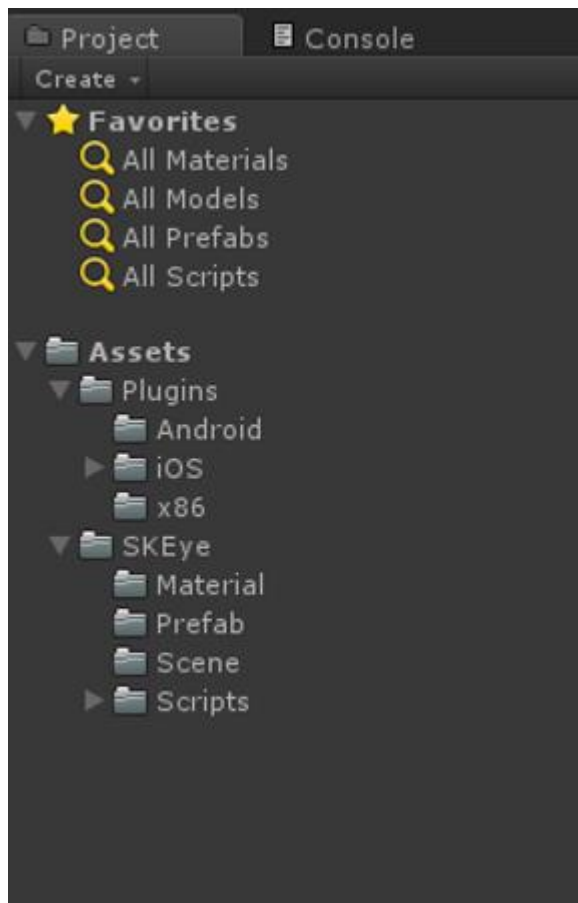
支持版本 Unity5.2.0 以上 建议使用 Unity5.2.0

3. 使用步骤

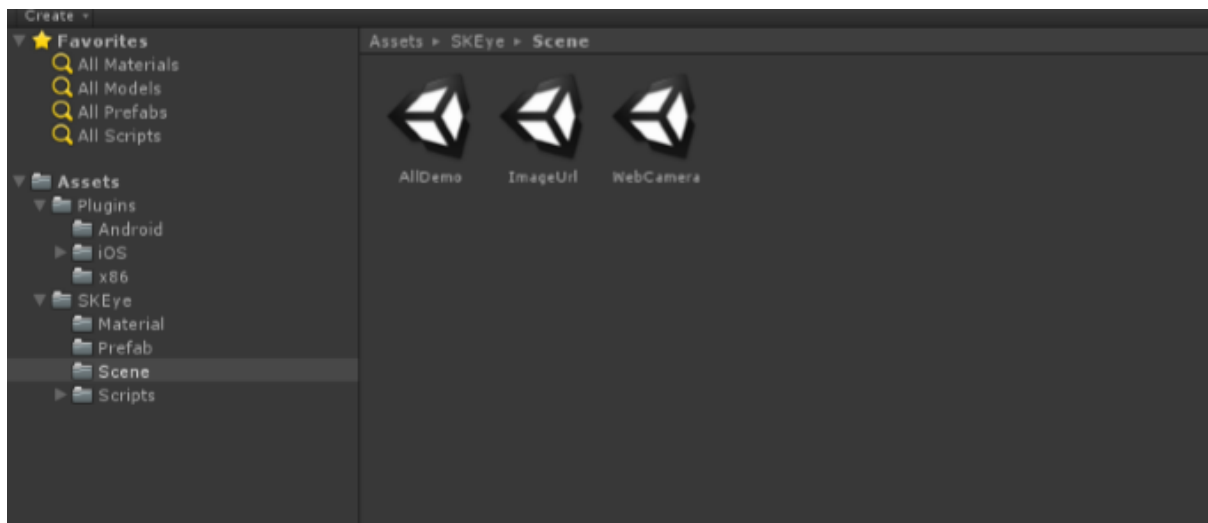
1) 首先安装 Unity3d, 建议使用 Unity5.2.0 (64 位)

2) 打开 Unity3d , 新建一个工程, 右键点击 Assets, 选择 Import Package , 继续选择 Custom Package. 选择 SKEye-Unity-SDK.unitypackage. (注意存放 unitypackage 的路径请不要有中文否则会导入不成功)

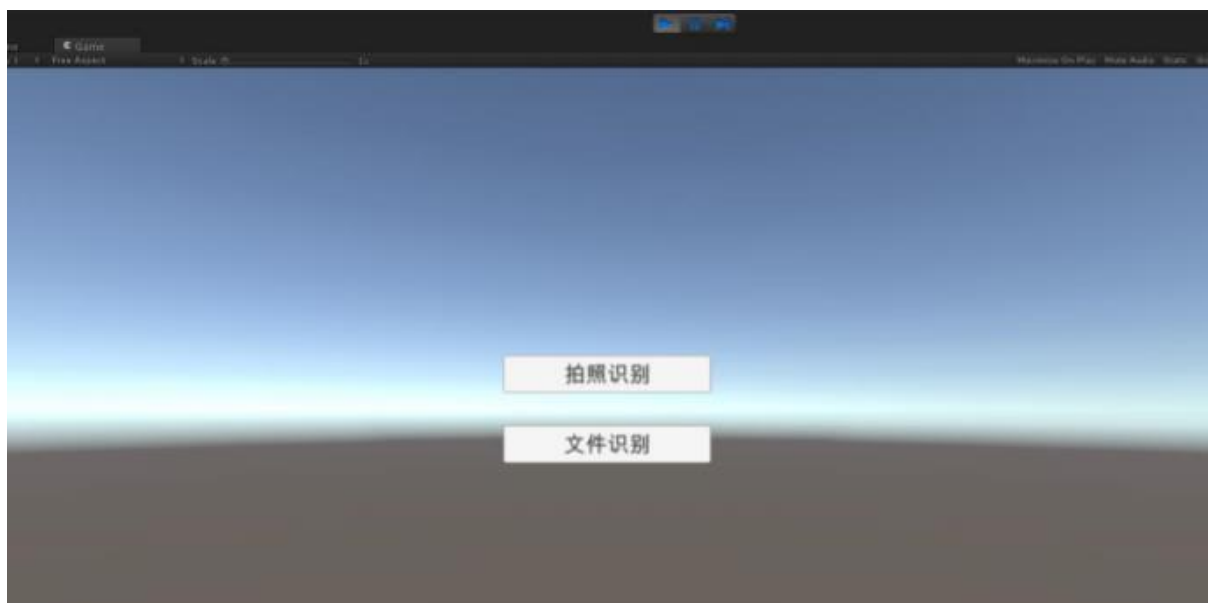
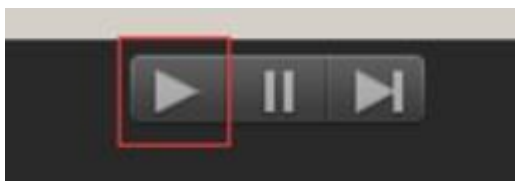
3) 导入 package 完毕后, Assets 下面的结构如下:



4) 选中 Assets/SKEye/Scene, 文件里存放了 Vu1.0.0 版本的所有 demo. WebCamera ,ImageUrl



5) 双击 AllDemo, 即可打开所有 Demo 的入口场景。点击开始按钮, 即可运行 DEMO 程序。 效果如下



5.1 WebCamera:拍照识别, 通过摄像头采集物体图像, 进行精准识别。

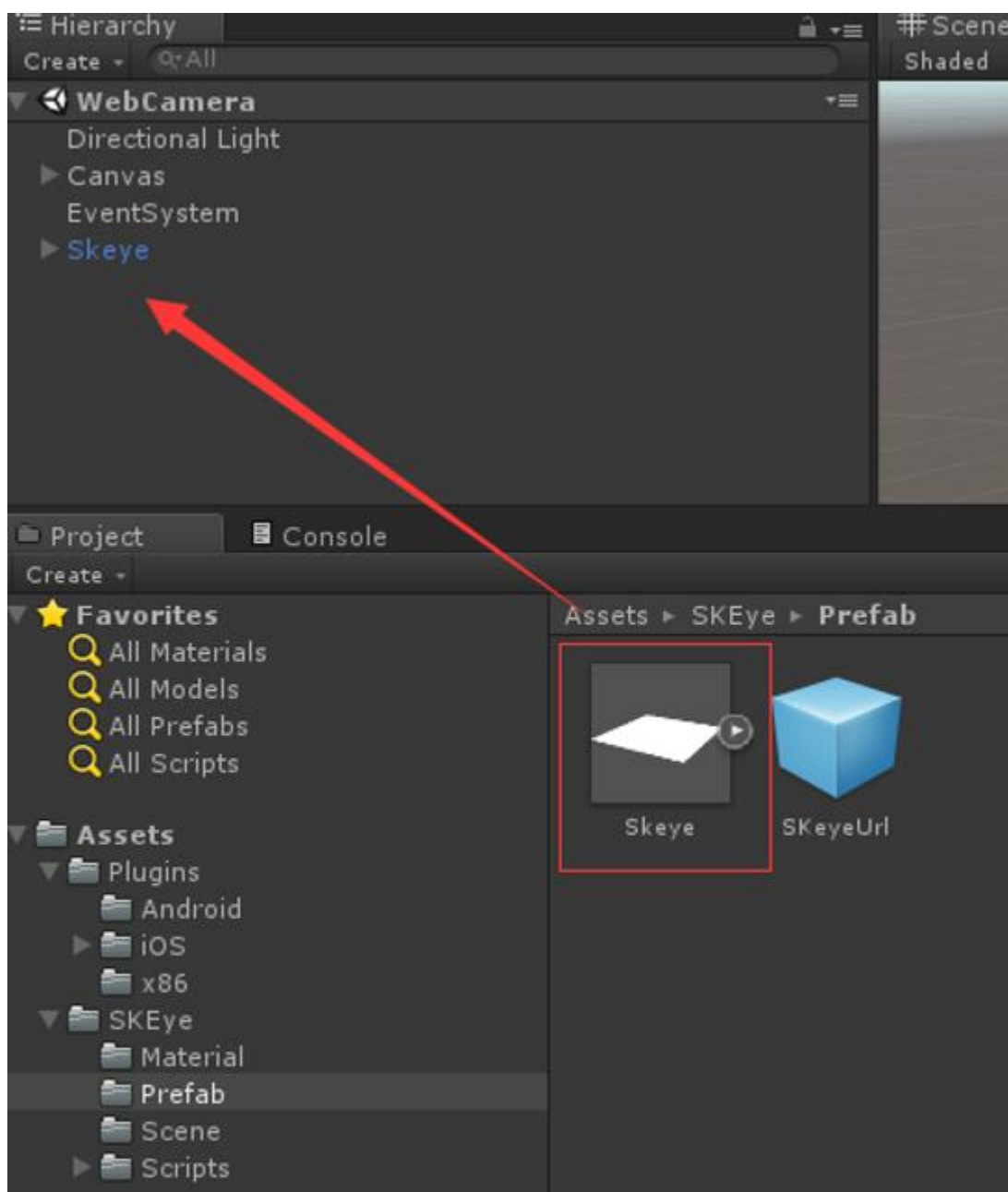
5.2 ImageUrl:路径识别, 如果图片路径(分为网络路径与本地路径)获取图片, 进行精准识别。

6) Demo 可以正常运行后, 用户可参考示例实现自己的场景。方式如下:

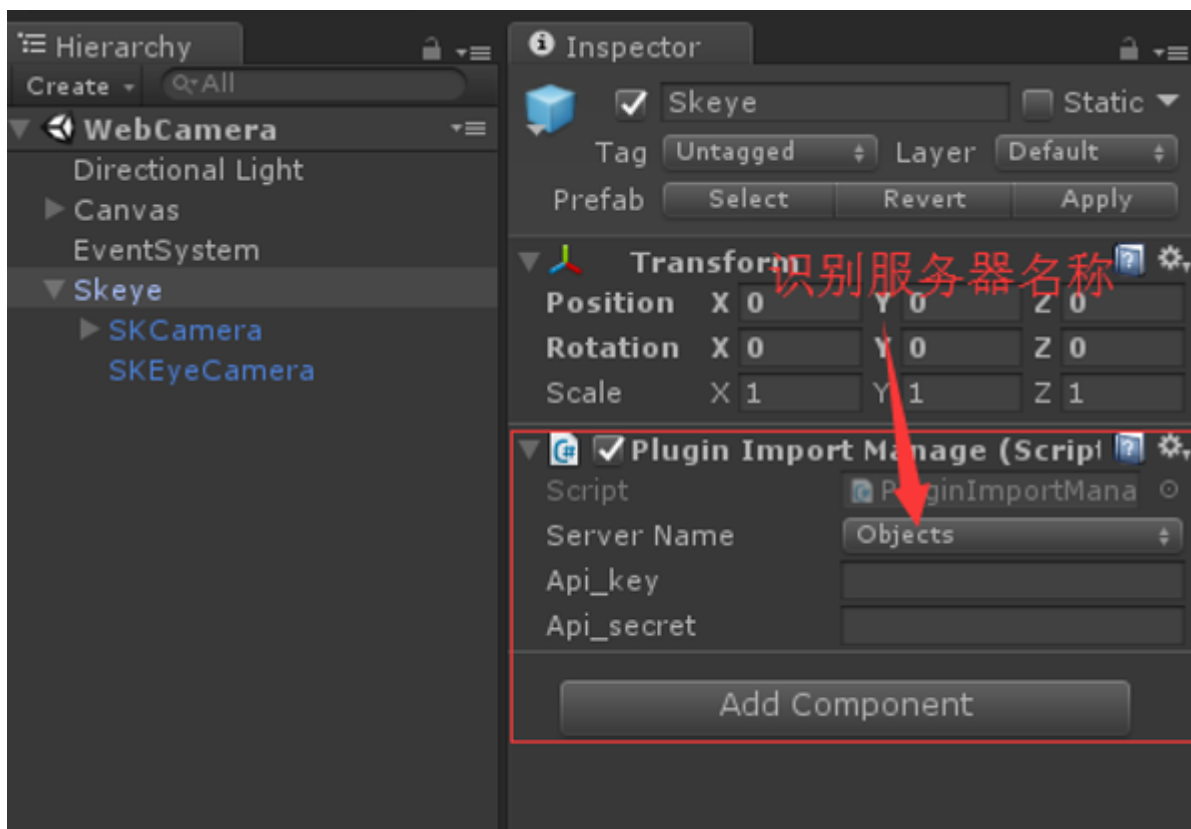
6.1: 创建一个新的 Scene, 如图



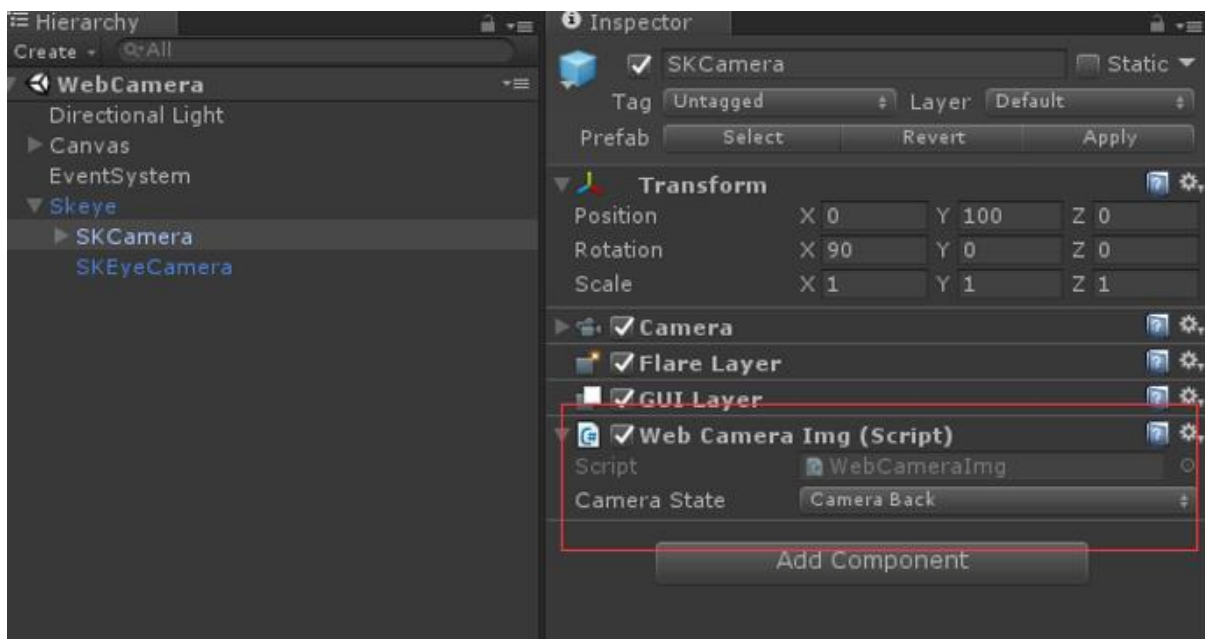
6.2: 选中 Assets/SKEye/Prefab 文件夹, 将 Skeye 预制件拖入到场景中, 如图:



6.3: 点击场景中的 Skeye, 查看其 Inspector 面板, 需要填写 Api_Key 以及 Api_secret, 如图:



6.4: 点击场景中的 SKCamera, 查看其 Inspector 面板, 在此设置前后摄像头, 如图



6.5: 完成以上操作后, 可以调用接口, 完成物体识别

4. Demo 说明

演示案例是 Unity 工程，案例展示如何调用 SKEyeSDK 接口，图像识别结果会在案例界面上展示，识别结果在输出界面上显示。

5. 关键参数说明

api_key: String 类型，指当前 APP 所申请的 SKEye 的 API_KEY
api_secret: String 类型，指当前 APP 所申请的 SKEye 的 API_SECRET
service_name: String 类型，指 SKEye 识别服务名称
float color_A: 红色 R 值偏色率
float color_G: 绿色 G 值偏色率
float bRateValue: 蓝色 B 值偏色率

6. 接口说明

6.1 Android SDK 初始化

```
public bool Initialise(string key, string secret)
{
    GetJavaClass().Call("SKEyeSDKInit", key, secret, SystemInfo.deviceUniqueIdentifier);
    return true;
}
```

参数:

- Verson : 当前版本
- api_key: 当前 APP 所申请的 SKEye 的 API_KEY
- api_secret: 当前 APP 所申请的 SKEye 的 API_SECRET

6.2 IOS SDK 初始化

```
public bool Initialise(string key, string secret)
{
    SKEyeSDK_InitWithApiKey(key, secret);
    return true;
}
```

参数:

- Verson : 当前版本
- api_key: 当前 APP 所申请的 SKEye API KEY
- api_secret: 当前 APP 所申请的 SKEye API SECRET

6.3 Windows SDK 初始化

```
public bool Initialise(string key, string secret)
{
    SKEyeSDK_Init(Encoding.Default.GetBytes(key),
        Encoding.Default.GetBytes(secret));
    return true;
}
```

参数:

- Version : 当前版本
- api_key: 当前 APP 所申请的 SKEye API KEY
- api_secret: 当前 APP 所申请的 SKEye API SECRET

6.4 图像偏色处理（调整图像偏色：偏红、偏绿、偏蓝）

适用情况：摄像机采集图像出现偏色现象。

```
public void SKEyeSDKColorCast(float color_A, float color_G, float color_B)
```

参数:

Color_A : 红色 R 值偏色率

Color_G : 绿色 G 值偏色率

Color_B : 蓝色 B 值偏色率

（偏色率计算公式：当前图片合适的 RGB 值，如果某个颜色值不需要改变，则传入 1，默认值为 1，则不对该颜色值做处理，R、G、B 颜色范围值：0~255）

备注:

如需使用该接口，请在 SDK 初始化后调用一次即可。如果图像不需要偏色处理，则不需要调用该接口

```
skEyeSDK.SKEyeSDKColorCast(1, (float)0.9, 1);
```

6.5 调用图像识别接口

(1) 拍摄识别

```
PluginImportManage.Instance.SKEyeSDKPhotograph();
```

可以参考 CameraRecognition.cs

(2) 图片路径识别

```
PluginImportManage.Instance.SKEyeSDKRoute(URL);
```

可以参考 RouteRecognition.cs

(3) 识别结果接收

```
void Start()    {
    PluginImportManage.Instance.MouseOver += Listener;
}

void Listener(string _ResurtStr)    {
    Debug.Log("识别结果 =" + _ResurtStr);
}
```

注：结果为 [Json 格式的字符串](#)

4. 开启摄像头

```
WebCameraImg.Instance.PlayWebCamera();
```

5. 关闭摄像头

```
WebCameraImg.Instance.StopWebCamera();
```

JSON 格式返回值：

```
{
  "error code": "0",
  "error msg": "OK",
  "tags": [
    {
      "tag name": "运动鞋",
      "tag_confidence": 81
    },
    {
      "tag name": "跑鞋",
      "tag_confidence": 18
    }
  ]
}
```

7. 补充

如果遇到问题，请按以下步骤尝试解决：

- 查看 [SKEye 人工智能开放平台 API 文档](#)
- 仔细阅读 Demo 接口调用源码
- 联系我们的工作人员
 - 客服电话：010-62538800
 - 技术支持邮箱：support@interjoy.com.cn
 - 开发平台用户 QQ 群：617518775