

SKEye-Android-SDK 说明文档

1. 概述

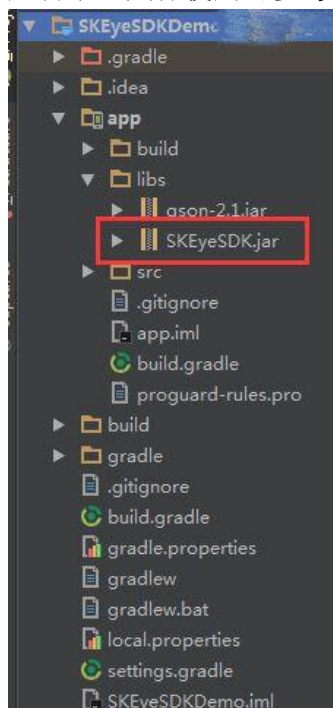
[SKEye](#) 是[盛开互动](#)依托十余年研发积淀推出的图像识别技术平台，致力于为智能产品、应用提供精准快速的“云+端”图像识别 SDK。SKEyeSDK 是一套基于 SKEye 人工智能开放平台的图像识别服务，该文档将详细说明如何在 Android 平台上快速集成该 SDK，并通过调用 SDK 接口使用 SKEye 图像识别服务。

2. 系统需求

Android4.0 及以上版本系统

3. 使用步骤

支持使用 Android Studio、Eclipse 进行集成使用，在新建工程后，将 SDK 拷贝至对应目录，具体使用可参考 Demo 工程。



4. Demo 说明

演示案例是 Android Studio 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，并以 log 形式打印。

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

ConstConfig.SKEyeSDK_SERVICE_NAME_OBJECT: String 类型, 指常见物品识别服务名称
ConstConfig.SKEyeSDK_SERVICE_NAME_FRUITS: String 类型, 指水果识别服务名称
bitmap: Bitmap 类型, 指 Bitmap 图片
image_url: String 类型, 指图片的网络 url/本地路径
image_YUVData: String 类型, 指图像的 YUV 数据 byte 数组
image_width: int 类型, 指图片的宽度
image_height: int 类型, 指图片的高度
rRateValue: float 类型, 指图像红色偏色率
gRateValue: float 类型, 指图像绿色偏色率
bRateValue: float 类型, 指图像蓝色偏色率

6. 接口说明

6.1 SDK 初始化

```
public SKEyeSDK(Context context);  
public void SKEyeSDKInit(String api_key, String api_secret);
```

参数:

- context: Android 上下文对象
- 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

示例代码:

```
SKEyeSDK skEyeSDK = new SKEyeSDK(getApplicationContext());  
skEyeSDK.SKEyeSDKInit("api_key", "api_secret");
```

6.2 图像偏色处理（调整图像偏色：偏红、偏绿、偏蓝）

```
public void SKEyeSDKColorCast(float rRateValue,  
float gRateValue, float bRateValue);
```

参数:

- rRateValue: 红色 R 值偏色率
- gRateValue: 绿色 G 值偏色率
- bRateValue: 蓝色 B 值偏色率

（偏色率计算公式：当前图片合适的 RGB 值/255（float 类型），如果某个颜色值不需改变，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。

示例代码:

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

6.3 调用图像识别接口

(1) 识别一个 Bitmap 图像

①直接调用

```
public String SKEyeSDK_Image(String service_name,
    Bitmap bitmap) throws IOException, JSONException,
    KeyManagementException, NoSuchAlgorithmException;
```

- 参数：
 - service_name: 识别服务名称
 - bitmap: Bitmap 图片
- 返回值：
[Json 格式的字符串](#)

②接口回调

```
public void SKEyeSDK_Image(String service_name,
    Bitmap bitmap, final ImageCallback imageCallback)
    throws IOException, JSONException,
    KeyManagementException, NoSuchAlgorithmException;
```

- 参数：
 - service_name: 识别服务名称
 - bitmap: Bitmap 图片
 - imageCallback: 回调接口
- 接口回调返回值：
[Json 格式的字符串](#)

示例代码:

```
// 直接调用
String result = skEyeSDK.SKEyeSDK_Image(
    service_name, bmp);
// 接口回调
skEyeSDK.SKEyeSDK_Image(
    service_name, bmp, new ImageCallback() {
    @Override
    public void recognitionInfo(String result) {
        // 在这里可以对识别结果result进行处理
    }
});
```

(2) 识别一个 url (网络 url、本地路径) 图像

①直接调用

```
public String SKEyeSDK_Image(String service_name,
    String image_url)throws IOException, JSONException,
    KeyManagementException, NoSuchAlgorithmException;
```

- 参数：
 - service_name: 识别服务名称
 - imageUrl: 图像的 url (网络 url 或者本地路径)
- 返回值：
[Json 格式的字符串](#)

②接口回调

```
public void SKEyeSDK_Image(String service_name,
    String image_url, final ImageCallback imageCallback)
    throws IOException, JSONException,
    KeyManagementException;
```

- 参数：
 - service_name: 识别服务名称
 - imageUrl: 图像的 url (网络 url 或者本地路径)
 - imageCallback: 回调接口
- 接口回调返回值：
[Json 格式的字符串](#)

示例代码:

```
// 直接调用
String result = skEyeSDK.SKEyeSDK_Image(
    service_name, image_url);
// 接口回调
skEyeSDK.SKEyeSDK_Image(service_name,
    image_url, new ImageCallback() {
    @Override
    public void recognitionInfo(String result) {
        // 在这里可以对识别结果result进行处理
    }
});
```

(3) 识别一个 YUV 数据格式图像

①直接调用

```
public String SKEyeSDK_Image(String service_name,
    byte[] image_YUVData,int image_width,
```

```
int image_height) throws IOException, JSONException,
KeyManagementException, NoSuchAlgorithmException;
```

- **参数:**
 - service_name: 识别服务名称
 - imageYUVData: 图像的 YUV 数据 byte 数组
 - imageWidth: 图像宽度
 - imageHeight: 图像高度
- **返回值:**
[Json 格式的字符串](#)

②接口回调

```
public void SKEyeSDK_Image(String service_name,
byte[] image_YUVData, int image_width,
int image_height, final ImageCallback imageCallback)
throws IOException, JSONException,
KeyManagementException, NoSuchAlgorithmException;
```

- **参数:**
 - service_name: 识别服务名称
 - imageYUVData: 图像的 YUV 数据 byte 数组
 - imageWidth: 图像宽度
 - imageHeight: 图像高度
 - imageCallback: 回调接口
- **接口回调返回值:**
[Json 格式的字符串](#)

示例代码:

```
// 直接调用
String result = skEyeSDK.SKEyeSDK_Image(
    service_name, imageYUVData, bmpW, bmpH);
// 接口回调
skEyeSDK.SKEyeSDK_Image(service_name, image_YUVData,
    image_width, image_height, new ImageCallback() {
    @Override
    public void recognitionInfo(String result) {
        // 在这里可以对识别结果result进行处理
    }
});
```

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