基于深度学习的多模态恶意图片检测系统

tuxiaobei/image_violation_detection - Docker Image | Docker Hub

接口说明

API `POST /check_img`

• 请求参数：

`base64` 与 `url` 参数二选一，分别表示图片的 `base64` 编码和图片的 `url` 地址，推荐使用 `base64`，从 `url` 抓取图片可能失败。

• 返回参数：

- `img`： 图片编号
- `white_result`： 白名单检测结果
- `black_result`： 黑名单检测结果
- `imgclass_result`：正常、涉黄、敏感图片的预测概率
- `malicious_detect`：二维码检测结果
- `sensitive_detect`：敏感词检测结果
- `ocr_result`：OCR 识别结果

• 返回样例：


```
{
    "code": 0,
    "data": {
        "img": "tmp/41d96af188e8640fe730bd2aebb344dd",
        "white_result":  {"filename": "a.png", "sim": 0.96},
        "black_result":  {"filename": "b.png", "sim": 0.52},
        "imgclass_result": [
            {
                "label": "normal",
                "prob": 0.9997885823249817
            },
            {
```

```
                "label": "porn",
                "prob": 0.00018990205717273057
            },
            {
                "label": "sensitive",
                "prob": 2.1459745767060667e-05
            }
        ],
        "malicious_detect": [
            {
                "Wording": "该网站发布了违反国家相关法律规定的内容，已为您拦截。",
                "WordingTitle": "网站含有违规内容",
                "code": 201,
                "detect_time": "2023-04-14 16:41:02",
                "msg": "域名拦截",
                "url": "https://aaa.com"
            }
        ],
        "ocr_result": "测试",
        "sensitive_detect": {
            "category": "None",
            "keyword": "None"
        }
    },
    "msg": "success"
}
```


运行说明

环境变量：

- `ocr_type` = `baidu` 或 `easyocr`，表示 OCR 识别方法，`easyocr` 则为本地识别，`baidu` 则为调用百度 API 识别，需开通 **通用文字识别 API**。若不包含此环境变量则表示不进行 OCR 识别
- `ocr_client_id` =填写百度 API key（仅 OCR 方法为 baidu 时需要）
- `ocr_client_secret` =填写百度 API secret（仅 OCR 方法为 baidu 时需要）
- `sensitive_detect_type` = `remote` / `local`，表示敏感词检测方法，是本地词库匹配还是调用 API，若不设置此环境变量则不进行敏感词检测
- `sen_api_key` =仅敏感词检测为 `remote` 需要，请前往 **该网站** 获取 API key
- `mal_api_key` =二维码域名检测需要，若不设置此环境变量则不进行二维码域名检测，请前往 **该网站** 获取 API key
- `knn_query_k` =黑白名单最大返回个数，若不设置默认为 1，最大不超过 50

- **运行命令：**

`docker run --env-file env.list -p 5500:5500 --name image_violation_detection -v img_data:/app/img_data --restart=always tuxiaobei/image_violation_detection`

`env.list` 即为环境变量列表，格式为

```
key1=value1
key2=value2
...
```

宿主机 `img_data` 目录即存储黑白名单图片，其中 `img_data/white` 下存储白名单图片，其中 `img_data/black` 下存储白名单图片