

## 1.1 百度云车牌识别

### 1.1.1 任务 6-百度云车牌识别概述

支持识别中国大陆机动车蓝牌、黄牌（单双行）、绿牌、大型新能源（黄绿）、领使馆车牌、警牌、武警牌（单双行）、军牌（单双行）、港澳出入境车牌、农用车牌、民航车牌的地域编号和车牌号，并能同时识别图像中的多张车牌。

#### 车牌识别应用场景：

车辆进出场识别-自动识别车辆车牌信息，应用于停车场、小区、工厂等场景，实现无卡、无人的车辆进出场自动化、规范化管理，有效降低人力成本和通行卡证制作成本，大幅度提升管理效率。

交通违章检测-自动识别定位违章车辆信息，实时检测并记录交通违章行为，有效降低人力监控成本，提升管理效率。

#### 车牌识别特点：

多颜色识别-可识别多种颜色大陆机动车车牌，包括蓝色、绿色、黄色、白色等，自动检测并识别车牌号码、车牌颜色、车牌位置。

多车牌识别-支持识别同一张图片中的多张车牌，针对车牌占比过小情况做专项优化，提升在路侧、监控高拍场景下的识别准确率。

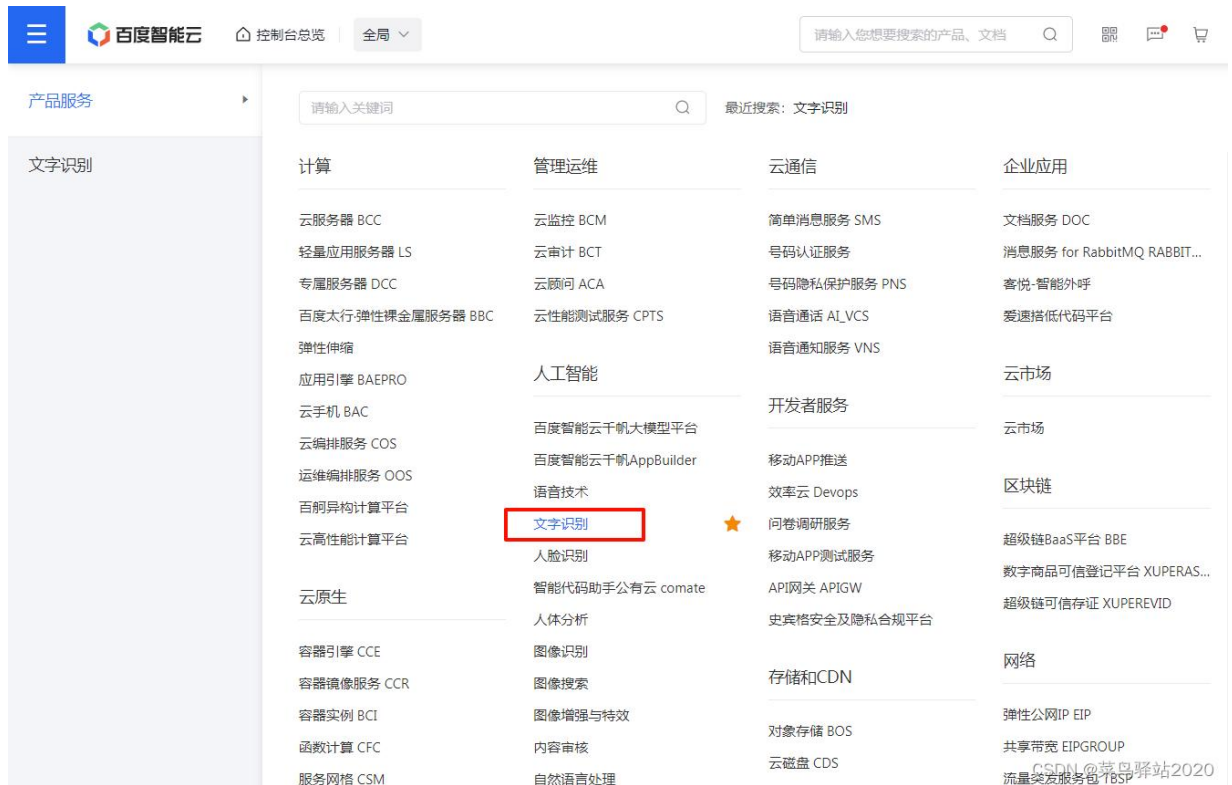
夜间车牌识别-支持夜间、弱光场景下的车牌识别，针对复杂光线、车牌反光等情况进行专项优化。

### 1.1.2 任务 7-注册百度智能云使用服务

注册百度智能云，选择产品服务。网址如下：

<https://console.bce.baidu.com/>

界面截图：



每天赠送 200 次，做开发测试足够了。



创建应用并在应用列表复制 AppID , API Key ,Secret Key 备用。



### 1.1.3 任务 8-百度云车牌识别在线工具

百度云官方车牌识别 **API** 调用教程链接：

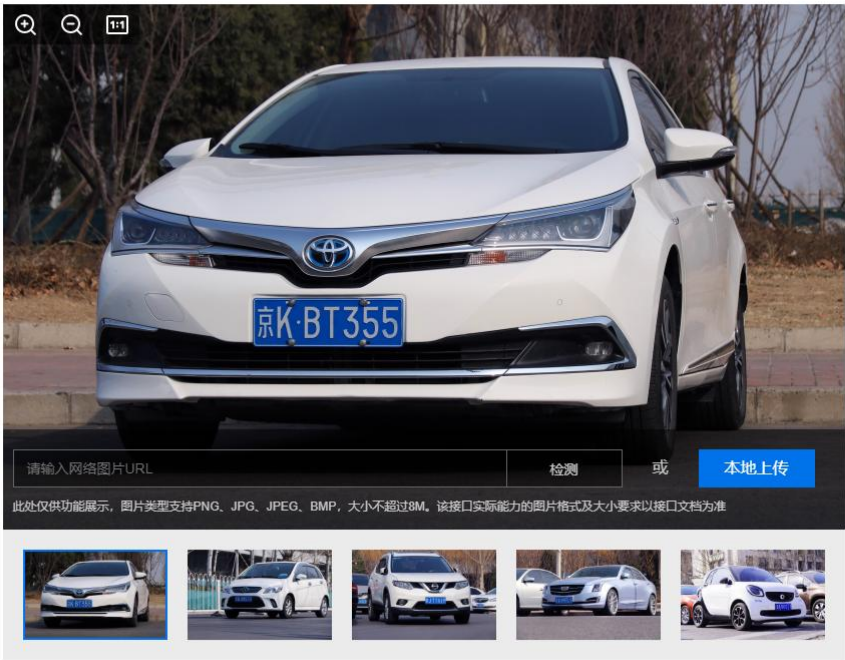
<https://cloud.baidu.com/video-center/video.html?id=741>



车牌识别功能演示网址：

[https://ai.baidu.com/tech/ocr\\_cars/plate](https://ai.baidu.com/tech/ocr_cars/plate)

功能演示



识别结果 请求参数 JSON结果

序号	内容
1	京KBT355 蓝牌

若识别有误，请点击反馈

在线调试：

您可以在 **示例代码中心** 中调试该接口，可进行签名验证、查看在线调用的请求内容和返回结果、示例代码的自动生成。

示例代码中心网址：

```
https://console.bce.baidu.com/tools/?_=1668473684721#/api?product=AI&project=%E6%96%87%E5%AD%97%E8%AF%86%E5%88%AB&parent=%E4%BA%A4%E9%80%9A%E5%9C%BA%E6%99%AFOCR&api=rest/2.0/ocr/v1/license_plate&method=post
```

服务与支持

大模型工具

大模型样板间

增值服务

自助专区

自助工具大厅

示例代码中心

智能助手

技术工单

创建工单

工单列表

授权列表

示例代码中心

大模型示例代码 人工智能 AI 基础云

文字识别 产品详情

请输入

鉴权认证机制

通用场景OCR

卡证OCR

交通场景OCR

行驶证识别

驾驶证识别

车辆证照混贴识别

车牌识别

车牌识别

鉴权参数 获取鉴权参数

ak\_sk access\_token

应用列表: ①

license\_plate

\* client\_id: ②

nZpAYT\*\*\*\*GUQARQ

\* client\_secret: ③

RQRlOx\*\*\*\*6M0q7T

测试

技术文档 示例代码 调试结果 教学视频

Python Java PHP C# Go Node.js C++

复制 下载 反馈

import requests

API\_KEY = "nZpAYT\*\*\*\*GUQARQ"

SECRET\_KEY = "RQRlOx\*\*\*\*6M0q7T"

def main():

url = "https://aip.baidubce.com/rest/2.0/ocr/v1/license\_plate?access\_token=" + g

payload='multi\_detect=false&multi\_scale=false'

headers = {

### 1.1.4 任务 9-Python 对接百度云车牌识别

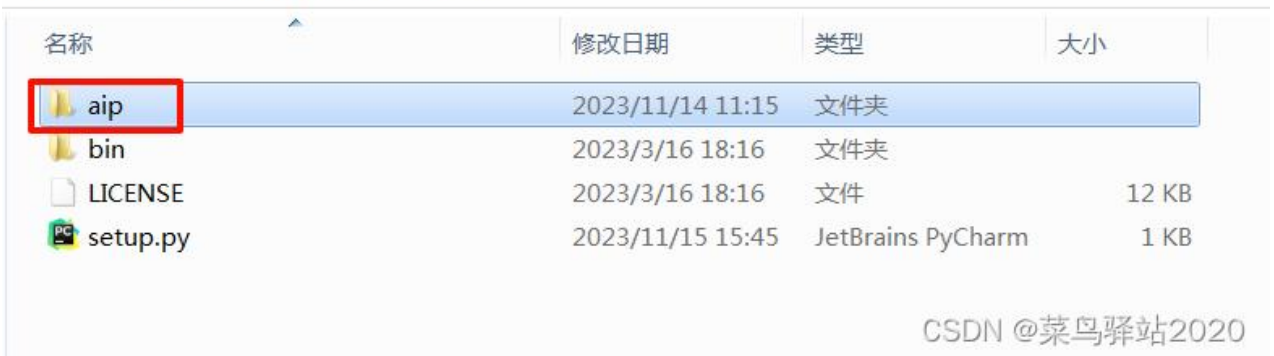
SDK 下载地址:

<https://ai.baidu.com/sdk#ocr>

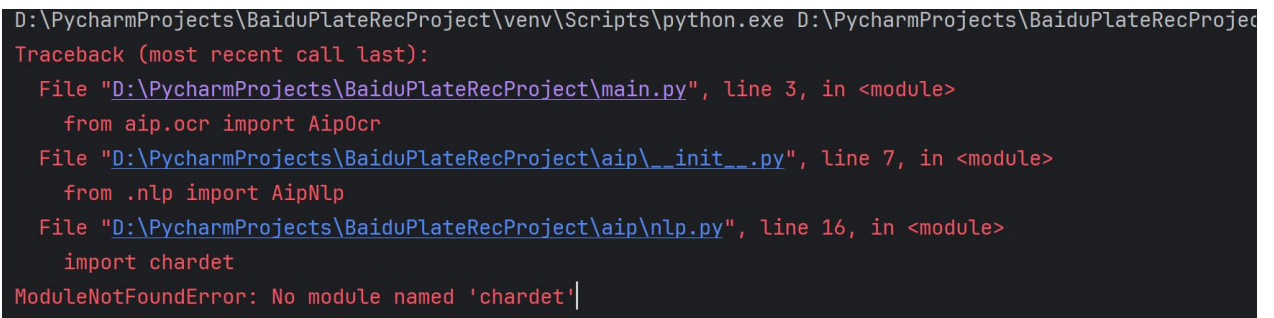


下载 SDK 文件, 解压, 复制 aip 文件夹到项目的根目录下备用。

库 > 下载 > 下载 > aip-python-sdk-4.16.13 >



但在使用的过程中, 可能还会缺少 `chardet` 库, 会抛出如下错误:



安装 `chardet` 库命令如下:

`pip install chardet`

利用我们前面创建的应用信息, 对接部分代码如下:

```
# 包的本质上是一个包含了一个或多个模块的文件夹，其中还包含一个特殊的文件__init__.py
```

```
# 要注意引入第三方的包的路径：包名.模块名 =》 aip.ocr
```

```
from aip.ocr import AipOcr
```

```
from PIL import Image, ImageDraw, ImageFont
```

```
import numpy as np
```

```
import cv2
```

```
# 修改成自己的
```

```
APP_ID = 'xxxxxxxx'
```

```
API_KEY = 'xxxxxxxxxxxxxx'
```

```
SECRET_KEY = 'xxxxxxxxxxxxxxxxxxxxxx'
```

```
# 创建客户端对象
```

```
client = AipOcr(APP_ID, API_KEY, SECRET_KEY)
```

```
# 建立连接的超时时间，单位为毫秒
```

```
client.setConnectionTimeoutInMillis(5000)
```

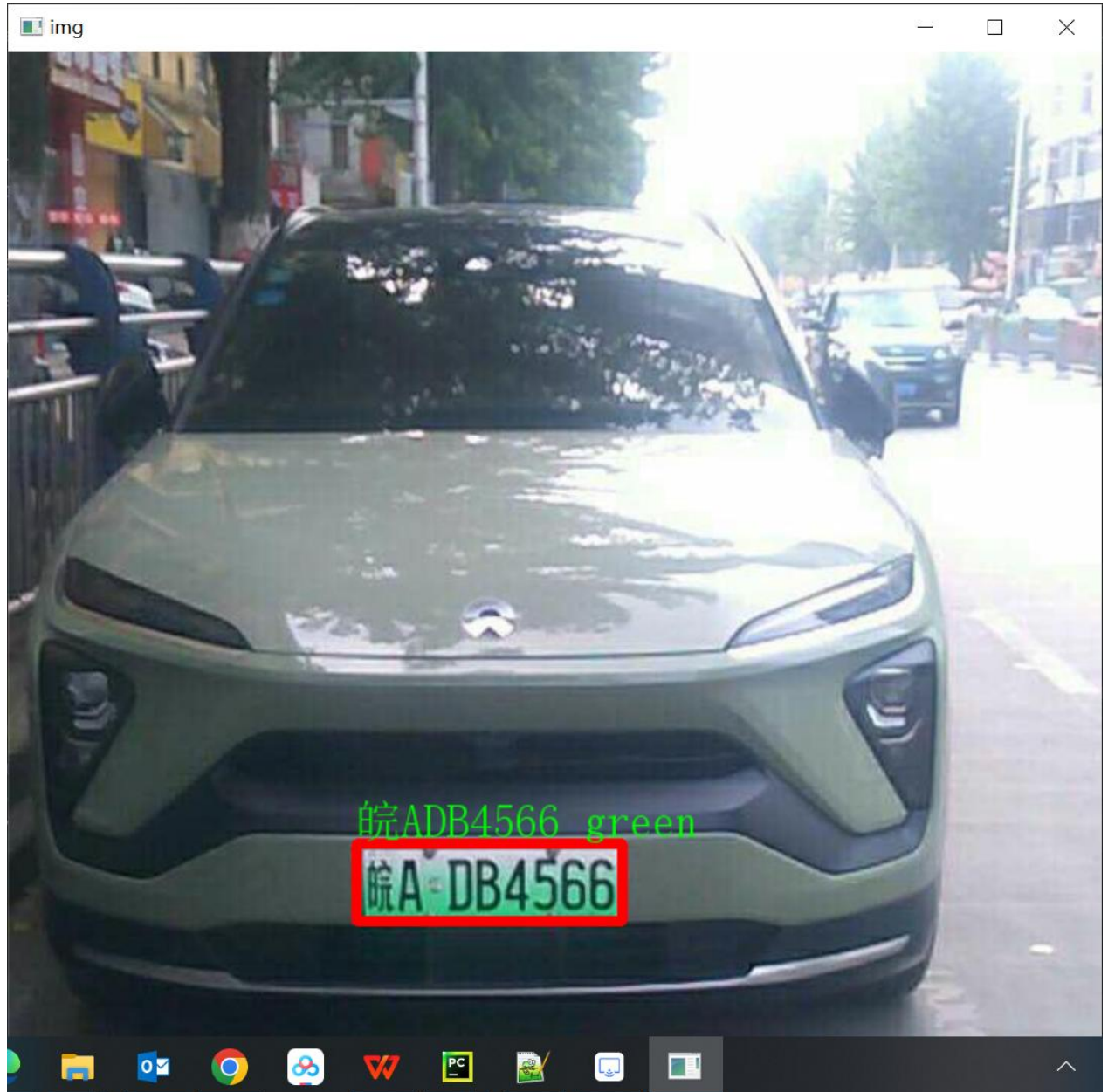
```
# 通过打开的连接传输数据的超时时间，单位为毫秒
```

```
client.setSocketTimeoutInMillis(5000)
```

### 1.1.5 任务 10-实现车牌识别并绘制到图片

该任务实现的效果如下：





完整实现代码如下：

```
# 包的本质上是一个包含了一个或多个模块的文件夹，其中还包含一个特殊的文件__init__.py

# 要注意引入第三方的包的路径：包名.模块名 =》 aip.ocr

from aip.ocr import AipOcr

from PIL import Image, ImageDraw, ImageFont

import numpy as np
```

```
import cv2

# 修改成自己的
APP_ID = '28222257'
API_KEY = 'YGMghwNb5VMfA5GAMC8NRY5E'
SECRET_KEY = 'LgzjKj75E3TlcjCwpXEEwrT1P2yboSc4'

# 创建客户端对象
client = AipOcr(APP_ID, API_KEY, SECRET_KEY)

# 建立连接的超时时间，单位为毫秒
client.setConnectionTimeoutInMillis(5000)

# 通过打开的连接传输数据的超时时间，单位为毫秒
client.setSocketTimeoutInMillis(5000)

""" 解决中文乱码 """
def cv2ImgAddText(img, text, left, top, textColor, textSize):
    # 判断是否 OpenCV 图片类型
    if (isinstance(img, np.ndarray)):
        img = Image.fromarray(cv2.cvtColor(img, cv2.COLOR_BGR2RGB))
    # 创建一个可以在给定图像上绘图的对象
    draw = ImageDraw.Draw(img)
    # 字体的格式
    fontStyle = ImageFont.truetype("simsun.ttc", textSize, encoding="utf-8")
    # 绘制文本
    draw.text((left, top), text, textColor, font=fontStyle)
    # 转换回 OpenCV 格式
    return cv2.cvtColor(np.asarray(img), cv2.COLOR_RGB2BGR)
```



```
""" 读取文件内容 """
```

```
def getFileContent(filePath):
```

```
    with open(filePath, 'rb') as fp:
```

```
        return fp.read()
```

```
""" 在图片上画出方框并写上文字 """
```

```
def imgDraw(imgPath):
```

```
    image_content = getFileContent(imgPath)
```

```
    # 百度返回的结果
```

```
    res = client.licensePlate(image_content)
```

```
    if res is not None:
```

```
        car_number = res['words_result']['number']
```

```
        car_color = res['words_result']['color']
```

```
        print('车牌号码: ' + car_number)
```

```
        print('车牌颜色: ' + car_color)
```

```
        # 车牌位置信息
```

```
        location = res['words_result']['vertexes_location']
```

```
        start_x = location[0]['x']
```

```
        start_y = location[0]['y']
```

```
        end_x = location[2]['x']
```

```
        end_y = location[2]['y']
```

```
        # 读取图片
```

```
        img = cv2.imread(imgPath)
```

```
        # 画长方形(图像矩阵, 左上角坐标, 右下角坐标, 颜色, 线条粗细)
```

```

cv2.rectangle(img, (start_x, start_y), (end_x, end_y), (0, 0, 255), 5)

# 在图片上绘制文字

txt = car_number + ' ' + car_color

# cv2ImgAddText(图像, 文字内容, 字体左边开始位置, 字体上面开始位置,
(R, G, B), 字体大小)
img_txt = cv2ImgAddText(img, txt, start_x+50, start_x+100, (0, 255, 0),
30)

# 显示效果

cv2.imshow('img', img_txt)

# 等待按任意键关闭窗口

cv2.waitKey(0)

cv2.destroyAllWindows()

else:

    print('车牌识别失败！')

if __name__ == '__main__':

    # 图片路径

    imgPath = 'plate.png'

    imgDraw(imgPath)

```

在代码中要注意引入百度云车牌识别 SDK 包的路径，否则会报错：  
ModuleNotFoundError: No module named 'ocr' 或者 ImportError: attempted relative  
import with no known parent package。截图如下：

