

日志@十七蝉

借助文字，一个人可以向众人说话，死者可以向生者说话，生者可以向未生者说话。

- [首页](#)
- [文章](#)
- [存档](#)
- [关于](#)
- [RSS](#)

[2014-03-21](#)

怎样在Android下保存OpenCV矩阵

如果只是在桌面系统（Mac、Windows或者Linux）使用OpenCV，OpenCV提供了将Mat保存到文件系统和从文件系统恢复Mat的C++ API。但是，在Android并未提供多这个API的Java API wrapper。

参考[Best way to store a Mat object in Android](#)，有以下方法：

1. 可能在Android下可以调用这个C++ API，这个需要找时间尝试
2. 有人提出了通过[gson](#)保存Mat的方法，应该是可用的。

我在实际项目中选择的是第二种办法，即通过java的gson实现的方式，不过他的代码在保存特殊矩阵的时候存在错误，比如直方图，会报告类型兼容错误。下面给出一种针对直方图的解决办法。

通过gson实现的基本思路是将Mat生成JSON字符串。

Mat对象的生成需要这样几个数据：

1. 行，row
2. 列，col
3. 矩阵类型
4. 矩阵的数据

前三项很简单，主要麻烦在数据。如果是图形矩阵的数据，可以：

```
1 public int get(int row,  
2               int col,  
3               byte[] data)
```

一次性的将字节数据取出来，然后生成BASE64编码的字符串，就好保存了。

但是，矩阵如果是直方图，调用这个方法的时候，会出现异常，因为它的类型并不匹配。解决办法是使用另外的方法：

```
1 public double[] get(int row,  
2                   int col)
```

这个方法并不按照矩阵类型自动获取数据，而是返回指定行列的数据。如果针对一个1列64行的直方图矩阵，我们可以循环64次将数据分别取出来，然后拼接字符串保存起来。比如：

```
1  JSONObject obj = new JSONObject();  
2  
3  if (mat.isContinuous()) {  
4      StringBuilder builder = new StringBuilder();  
5  
6      for (int i = 0; i < mat.rows(); i++) {  
7          double[] d = mat.get(i, 0);  
8          builder.append(d[0]);  
9          if (i != mat.rows() - 1) {  
10             builder.append(",");  
11         }  
12     }  
13  
14     obj.addProperty("rows", mat.rows());  
15     obj.addProperty("cols", mat.cols());  
16     obj.addProperty("type", mat.type());  
17     obj.addProperty("data", builder.toString());  
18  
19     Gson gson = new Gson();  
20     String json = gson.toJson(obj);
```

```
21
22     Log.d("opencv_detector", "json: " + json);
```

这个字符串恢复成矩阵，代码类似这样：

```
1  JsonParser parser = new JsonParser();
2  JsonObject jsonObject = parser.parse(json).getAsJsonObject();
3
4  int rows = jsonObject.get("rows").getAsInt();
5  int cols = jsonObject.get("cols").getAsInt();
6  int type = jsonObject.get("type").getAsInt();
7
8  String dataString = jsonObject.get("data").AsString();
9
10 Mat mat = new Mat(rows, cols, type);
11
12 int rowIndex=0;
13
14 for (String s : dataString.split(",")) {
15     mat.put(rowIndex++,0,Double.parseDouble(s));
16 }
```

完整的代码，可见Gist：

[android](#), [opencv](#)

微信公众号



标签

- [AFNetworking](#)¹
- [Android](#)¹
- [BLE](#)³
- [BlueTooth](#)³
- [Browserify](#)¹
- [CSS3](#)²
- [Docker](#)⁵
- [EasyRTC](#)¹
- [Flux](#)¹
- [HTML5](#)⁴
- [Hexo](#)²
- [Mac](#)¹
- [OHHTTPStubs](#)¹
- [RSA](#)¹
- [React](#)⁵
- [SDWebimage](#)¹
- [Swift](#)⁴
- [Translation](#)¹
- [Ubuntu Server](#)¹
- [Unity](#)²
- [WeChat](#)¹
- [WebRTC](#)²
- [android](#)²
- [arp](#)¹
- [bower](#)¹
- [canjs](#)¹

- [crypto](#)¹
- [docker](#)¹
- [gulp](#)²
- [iOS](#)⁷
- [momentjs](#)¹
- [mqtt](#)¹
- [nodejs](#)³
- [numpy](#)¹
- [opencv](#)²
- [python](#)¹
- [qiniu](#)¹
- [qrcode](#)¹
- [requirejs](#)¹
- [rsync](#)¹
- [sfnttool](#)¹
- [stats](#)¹

© 2015 十七蝉

