

# 驾驶员状态检测

---



## 项目介绍

---

在这个项目中，我们会通过深度学习的方法来检测驾驶员的状态，判断驾驶员是否专心驾驶。

输入: 是一张RGB的图片，大小是 (640, 480, 3)

输出: 10 种状态的概率分布

所有的 10 种状态分别是：

- c0: 安全驾驶
- c1: 右手打字
- c2: 右手打电话
- c3: 左手打字
- c4: 左手打电话
- c5: 调收音机
- c6: 喝饮料

- c7: 那后面的东西
- c8: 整理头发和化妆
- c9: 和其他乘客说话

## 数据下载

---

数据可以通过 kaggle 的比赛 [Distracted Driver Detection](#) 进行下载

如果你无法访问 kaggle 的网站，可以通过[百度云](#)上进行下载

数据下载完成之后，将本仓库下载到本地，将 `imgs.zip` 和 `driver_imgs_list.csv.zip` 放在仓库的主目录中

打开终端，进入到仓库目录，运行 Python 脚本

```
python get_data.py
```

上面的数据解压和重新排列可能会需要一点时间

通过上面的过程，我们准备好了数据，如果你已经根据 [StartKit](#) 配置好了深度学习环境，那么你可以直接进入 `distracted_driver_detection.ipynb` 完成项目，否则你需要根据 StartKit 配置你的深度学习环境。

## 一些建议

---

notebook 中已经有了完整的项目指南，你也可以使用本章我们所讲的一些模型，比如VGG，ResNet，DenseNet 等等。这些模型在 PyTorch 的 [model\\_zoo](#) 中都有完整的实现，你可以直接对其进行[调用](#)，同时你也能够对模型进行一些自定义的修改。

## 评估与提交

---

通过 `distracted_driver_detection.ipynb`，你能够创建一个可以提交的csv文件，将这个文件提交到 kaggle 中获取分数，这个分数作为我们的评估结果。同时你还需要提交

- kaggle 上提交分数的截图
- 完整的 notebook 代码文件
- notebook 导出的 html 文件

可以考虑在 Github 上为该项目创建一个仓库，记录训练的过程、所使用的库以及数据等的 README 文档，构建一个完善的 Github 简历。