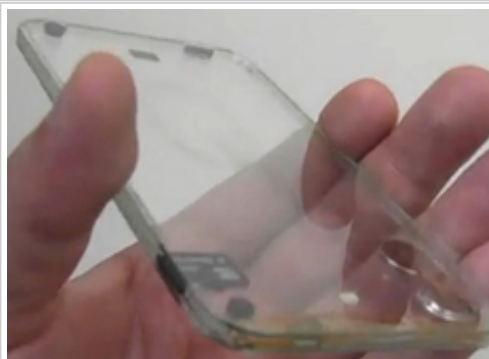
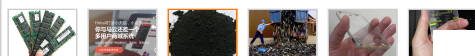




钨碳回收



透明手机



访问：5613次

积分：281

等级：**BLOG > 2**

排名：千里之外

原创：20篇

转载：24篇

译文：0篇

评论：2条

目录视图

摘要视图

RSS 订阅

图灵赠书——程序员11月书单 【思考】Python这么厉害的原因竟然是！ 感恩节赠书：《深度学习》等异
作译者评选启动！ 每周荐书：京东架构、Linux内核、Python全栈

MobileNet教程：用TensorFlow搭建在手机上运行的图像分类器

2017-08-01 16:56

980人阅读

评论(0)

收藏

分享

分类：

量子位 (2)

目录(?)

[+]

关闭

作为卷积神经网络中的新成员，MobileNet有着很多令人惊叹的特点，本文将带你训练一个试试。

MobileNet具有以下酷炫的特点：

1. 它们非常非常小
2. 它们非常非常快

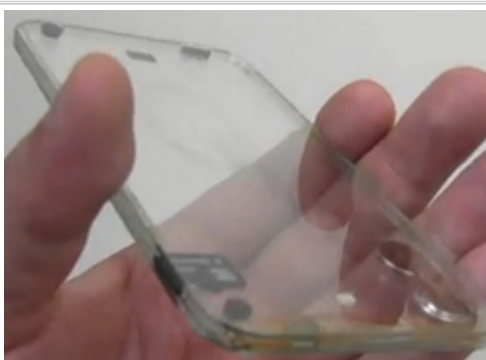
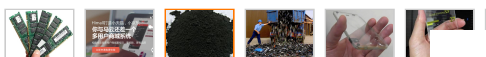


透明手机售价





钨碳回收



透明手机



matlib (7)

CSP (4)

大数据 (1)

文章存档

2017年10月 (15)

2017年09月 (1)

2017年08月 (19)

3. 它们非常非常准

4. 它们很容易调试

这些特点是非常重要的。

目前，很多移动端上的深度学习任务都是在云端完成的。当你想要让手机识别一张图片，程序会先把这张图片通过网络发送到远程服务器上进行分类，随后再将结果发送回手机上。

随着手机计算能力的迅猛增加，加上SqueezeNet和MobileNet等架构让计算机的网络复杂度快速下降，深度学习计算很快就能完全在设备本地完成。

移动设备本地的深度学习，除了能在没有网络连接的情况下正常运行之外，另一省时间，比如说一个车辆安全应用，对反应速度要求非常高，把图片传送到云端处理是不现实的。

因此，本文按照以下的顺序来介绍MobileNet：

1. MobileNet是什么？

2. 怎样搭建自己的数据集，在TensorFlow下训练MobileNet

3. 怎样用TensorFlow训练一个在ImageNet上训练过的模型

4. 跟Inception V3相比，MobileNet的表现怎么样？

5. 怎样使用再训练(retrained)的MobileNet来识别图片？

关闭

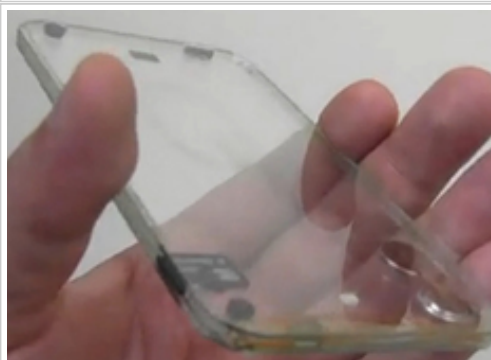
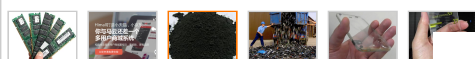


透明手机售价





钨碳回收



透明手机



测试数据科学家聚类技术的40... (0)

强化学习入门第一讲 马尔科夫... (0)

一文了解各种卷积结构原理及... (0)

Windows下安装Scikit-Learn (0)

面试题之走楼梯问题 (0)

数据缺失时，怎样选择合适的... (0)

最长公共子序列与最长公共子串 (0)



此图片来自微信公众平台

未经允许不可引用

MobileNets是什么？

MobileNet是由Google的研究者们设计的一类卷积神经网络。它们在手机上运行耗小、运行速度快，因此很适合在移动端上做应用。

MobileNet和传统的CNN在结构上的差别主要是，传统CNN中在批规范化和ReLU函数前边，是一个3×3卷积层，而MobileNet将卷积过程分为一个3×3深度分离卷积和一个1×1点对点的卷积。如果你想了解个中细节和缘由的话，我强烈建议你读一下这篇论文。

论文：

MobileNets: Efficient Convolutional Neural Networks

<https://arxiv.org/pdf/1704.04861.pdf>

那么MobileNet的短板是什么呢？准确性。跟我们熟悉的网络相比，MobileNet的准确性不如前者高。但是MobileNet寻求良好的平衡点。

MobileNet拥有两个表观变量：width multiplier和resolution multiplier。调整这两个变量值来使得模型适应具体问题。Width multiplier和resolution multiplier可以改变输入图片的分辨率，从而降

关闭



透明手机售价



Google开源了MobileNet，并随之开放了16个ImageNet checkpoint，每一个对应一种不同的参数结构。这为我们训练自己的小又快的图像分类器提供了一个良好的开端。

搭建数据集，训练MobileNet

我们今天的挑战是搭建一个能够识别道路和非道路图片的分类器。这就像《硅谷》里面的“hot dog, not hot dog”应用，把热狗改成了道路。

为什么选择道路呢？因为在Coastline，我们正在基于计算机视觉开发用于汽车安全的移动应用。跟所有涉及视觉的app一样，用户隐私是非常需要考虑的一点。所以当用我们的app时，系统会首先检查它看到的是否是道路。如果不是的话，那么它就会关闭我们希望这个过程能够在尽可能快速、只占用少量资源的情况下完成。

为了解决这个问题，我们需要先为它创建数据集。我们的目标是收集10,000张图片，道路和非道路的图片大概各占50%。

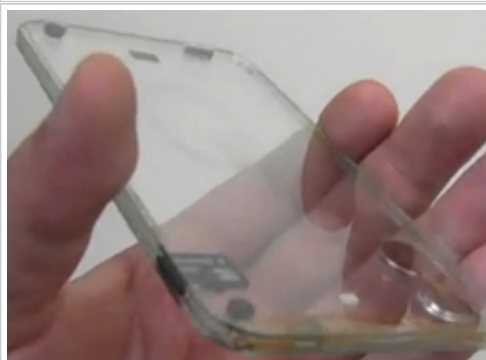
我们从以下几个来源得到相应的图片：

- 从我们自己的数据集里，随机选取4,000张显然是道路
- 从ImageNet里随机选取2000张显然不是道路的图片
- 从网上选取3000张不那么明显的非道路图片，以防分类“空”；
- 从网上选取1000张不那么明显的道路图片，以防分类为道路特征。

关闭



钨碳回收



透明手机

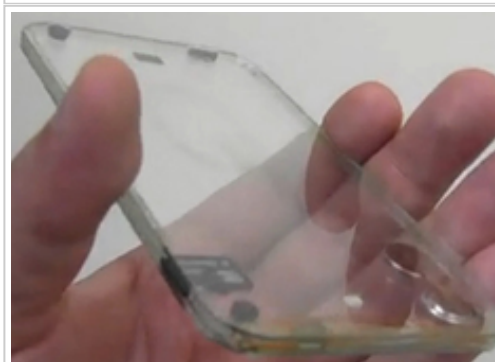
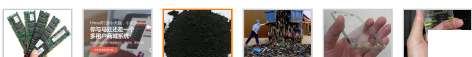


透明手机售价





钨碳回收



透明手机



我们将会把图片放进“道路”或“非道路”文件夹，这就是我们在重新训练网络之前所需要的图片上的准备工作。

此外，从网上搜集的图片可以有效地增加你的数据集的多样性，但这样做也有一个缺点，网站上图片的标签往往有些混乱。比如说，通过搜索“道路风景（road landscape）”所得到的图片可能是在美美的自然风光背景下郑重有条通向前方的道路：



或者是群山有一条隐约可见的小路：



为解决这个问题，我们可以挨个浏览每张图片然后手动进Deep learning干嘛呢？

实际上，我们将在所有数据的基础上重新训练一个大型的要通过early stopping和heavy data augmentation等方法



透明手机售价



关闭

然后我们让网络对数据集中所有的图片（包括之前用来训练的图片）进行分类，然后记录那些分错类或不确定的图片。随后我们逐个排查这些图片，把它们移动到正确的分类下。

这与之前手动分类图片相比，需要进行的调整的图片数量大大减少。重复这个多次步骤，能够让我们的准确率比Inception提高7个百分点。

随后我们将会使用TensorFlow以及迁移学习来在我们这个特定的数据集上对MobileNet进行调整。

使用ImageNet预训练过的模型

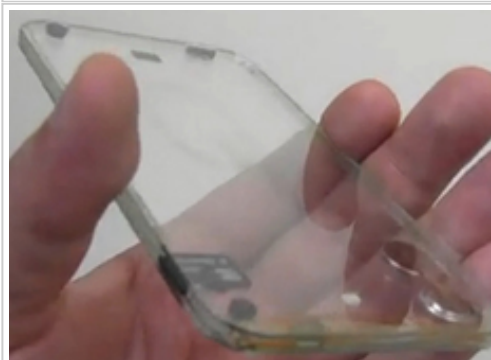
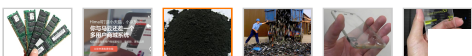
TensorFlow拥有一些很好的工具，你可以使用它们在不码任何代码的情况下就练MobileNet。

TensorFlow下再训练MobileNet详情：

<https://github.com/tensorflow/tensorflow/commit/055500bbcea60513c0160d212a10a7055f079312>



钨碳回收



透明手机



此图片来自微信公众平台
未经允许不可引用

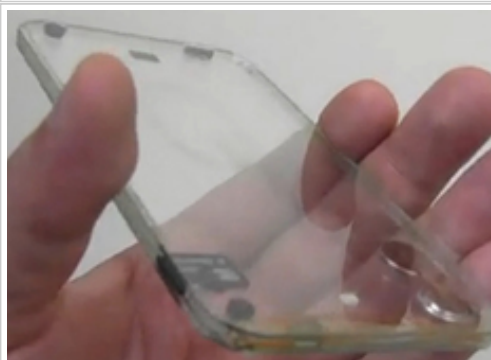
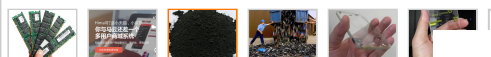


透明手机售价





钨碳回收



透明手机



此图片来自微信公众平台
未经允许不可引用

你可以用TensorFlow范例文件夹里的脚本文件，来在你自己的数据上重新训练MobileNet。

等等！你应该使用哪一版的MobileNet呢？这是个好问题。让我们先简单训练一版本的表现。为了开始训练，我们将会运行以下一段来自TensorFlow repository的脚本：

```
python tensorflow/examples/image_retraining/retrain.py \
  --image_dir ~/ml/blogs/road-not-road/data/ \
  --learning_rate=0.0001 \
  --testing_percentage=20 \
  --validation_percentage=20 \
  --train_batch_size=32 \
  --validation_batch_size=1 \
  --flip_left_right True \
  --random_scale=30 \
  --random_brightness=30 \
```

关闭

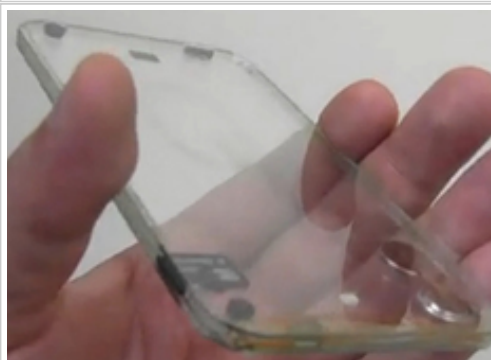


透明手机售价





钨碳回收



透明手机



```
--eval_step_interval=100 \
--how_many_training_steps=600 \
--architecture mobilenet_1.0_224
```

代码中“architecture”的标签对应着将要重新训练的MobileNet的版本。这里“1.0”对应着width multiplier，取值范围有1.0，0.75，0.50和0.25。“224”则对应着image resolution/resolution multiplier，这里可以选择的值有224，192，160和128。也就是说，为了训练最小型的MobileNet，你应该用“—architecture mobilenet_0.25”。

代码中所涉及的其他重要的参数有：

learning_rate：这是你需要进行调整的参数，我发现设置为0.0001有较好的训练效果。

testing and validation percentage：这个脚本将会把你的数据分为 训练集/验证集/测试集三类。网络将会使用训练集进行训练，在每一个“eval_step_interval”之后使用验证集对网络表现的评估进行更新，最后在“how_many_training_steps”之后使用测试集讲出最终的分数。

validation_batch_size：把这项设为-1意味着让脚本使用没有很多数据时（比如说这里我们只有10000张图片），步骤中的差异。

用这种方法训练几个版本的MobileNet之后，让我们看看

MobileNet表现如何？



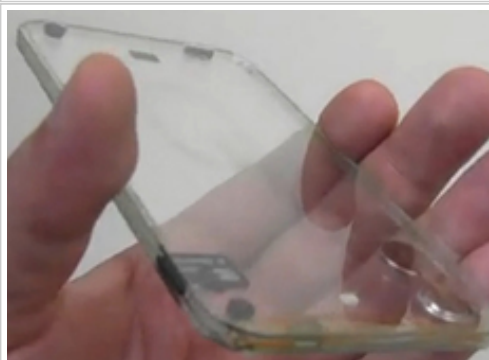
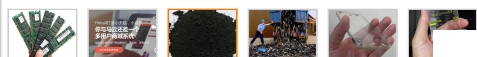
透明手机售价



关闭



钨碳回收



透明手机



为了得到比较的基准，我们花18分钟对Inception进行了600步训练，最终达到了95.9%的正确率，模型大小有84MB。

在测试对1,000张图片进行快速识别时发现，Inception可以在NVIDIA GeForce 960m GPU的架构上以19fps (frame per second，每秒钟识别的图片数)的速度对图片进行识别。

当然，这个结果还有很多改进空间，但我们不打算在这上面继续花时间了。

训练Inception V3，只需要将脚本中的-architecture-标签那里改为inception_v

旁注：为什么准确率只有95.9%？这看起来应该是个很好解决的问题。确实，除对训练参数进行充分的调试之外（我们实际上在另一次采用不同结构的训练中的准确率），其实这两个类别之间的界定也有一些模糊。比如说：

有一张图片的内容是是树林中一条不清晰的小径。这到底是一条铁路还是马路？我看不清楚。

有一张图片是，在风景名胜中的远处有条路。这到底是算是算作是一张图片中的道路？什么情况下我们认为这个分

有一些偏艺术风的图片，是否应该算作是道路呢？

那么，MobileNet的表现如何呢？不出意外，没那么好。但面有了很大的提升。

使用最大的MobileNet (1.0, 224)，我们能够在4分钟的时间内最终产生的模型大小也仅有17Mb，在同样的GPU下能够以1



透明手机售价



关闭

与Inception相比，MobileNet训练出的模型速度是前者的7倍，而尺寸只有前者的四分之一，准确率上只损失了0.4%。

那么最小尺寸的MobileNet（0.24, 128）表现怎么样呢？准确率下降了更多，只有89.2%，但是运行速度有450 fps，模型的大小仅有930kb，还不到1Mb！

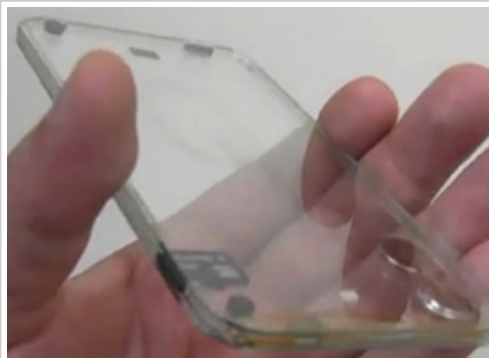
用再训练的MobileNet来分类

现在你拥有了在你特定数据集上的重新训练后的MobileNet，是时候来试试了。不出意外的是，TensorFlow上也有相应的脚本文件来完成这项功能。

```
python3 tensorflow/examples/label_image/label_image.py \
--graph=/tmp/mobilenet_0.50_192.pb \
--labels=/tmp/output_labels.txt \
--image=/home/harvitronix/ml/blogs/road-not-road/test-image.jpg \
--input_layer=input \
--output_layer=final_result \
--input_mean=128 \
--input_std=128 \
--input_width=192 \
--input_height=192
```

[关闭](#)


钚碳回收



透明手机

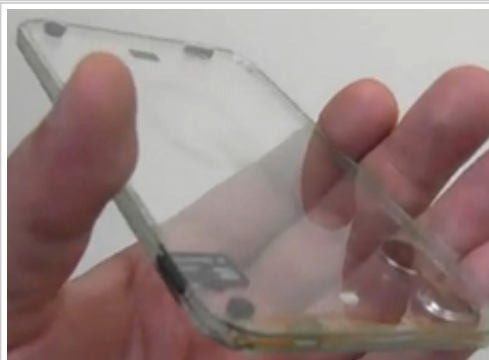


透明手机售价





钨碳回收



透明手机



此图片来自微信公众平台
未经允许不可引用

△ 我们的算法把这张图片识别为道路，虽然信心值只有0.686811，但也很不错了

旁注：值得说明的是，在我们这个相当简单的两分类问题中，准确度（与模型大小、运行速度之间）的权衡没有这么显著。对于像拥有1001个分类的ImageNet来说，这个非常重要了。

见Google公布的表格：

<https://research.googleblog.com/2017/06/mobilenets-open-source-models>

转自：微信公众号：量子位(QbitAI)

原文链接：<https://mp.weixin.qq.com/s/RxPhgKtPuNkUUn2G80ye4A>

关闭



透明手机售价



顶 踩
1 0

- 上一篇 深度学习动手入门：GitHub上四个超棒的TensorFlow开源项目
- 下一篇 年薪30万-50万，站在风口上的大数据面临150万的人才紧缺

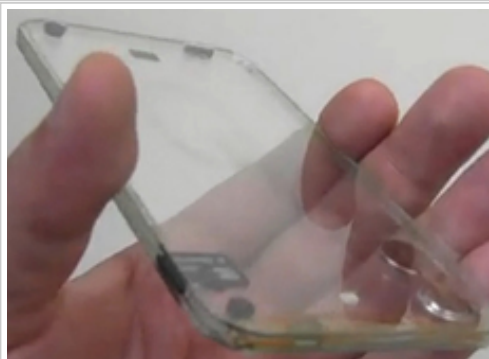
相关文章推荐

- TensorFlow MobileNet
- 腾讯云容器服务架构实现介绍--董晓杰
- TensorFlow object detection api-----ssd_mobilenet使用
- 容器技术在58同城的实践--姚远
- 谷歌开源移动端视觉识别模型：MobileNet
- Tensorflow项目实战-文本分类
- tensorflow开源Tensorflow Object Detection API安装...
- MySQL深入浅出
- 将 TensorFlow 移植到 Android手机，实现
- Python可以这样学（第三季：多线程与多
- 基于 TensorFlow 在手机端实现文档检测
- 华为工程师，带你实战C++
- 在Android上使用Tensorflow
- 让深度学习
- tensorflow
- building-m

关闭



钨碳回收



透明手机



零基础学习手绘



workflow管理系统



客户管理系统



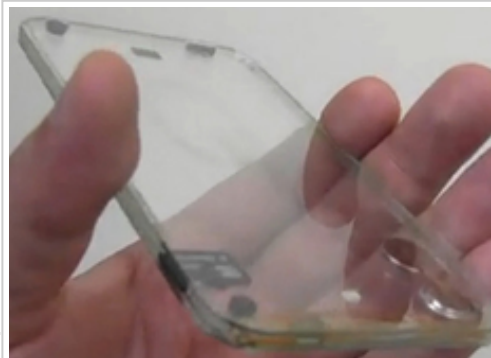
透明手机售价



查看评论



钨碳回收



透明手机



暂无评论

发表评论

用户名： weixin_35068028

评论内容：



提交

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

方式 | 版权声明 | 法律顾问 | 问题报告 | 合作伙伴 | 论坛反馈

关闭

webmaster@csdn.net 400-660-0108 | 北京创新乐知信息技术有限公司 版权所有 | 江苏知

京 ICP 证 09002463 号 | Copyright © 1999-2017, CSDN.NET, All Rights Reserved



透明手机售价

