

DeepCamera

安卓应用设计实验报告

易凯

Email: williamyi96@gmail.com

Project Page: <https://github.com/WilliamYi96/DeepCamera>

2017 年 6 月 28 日

Contents

1	关于本报告的说明	7
1.1	致谢	7
1.2	内容声明	8
1.3	作者信息	9
2	实验研究基本背景	11
2.1	基本介绍	11
2.2	TensorFlow	11
2.3	JNI-DNK	12
3	项目总体设计	15
3.1	总体流程图	15
3.2	总体设计描述	15
4	项目模块化设计	17
4.1	TensorFlow 模型 Android 应用	17
4.2	Image 分类识别实现	17
4.2.1	设置选择标准	17
4.2.2	将物品标签写入内存	17
4.2.3	将图像数据转换为 bitMap 支持的数据格式	17
4.2.4	将输入的数据导入 TensorFlow 模型	18
4.2.5	运行模型推理得到输出 tensor	18
4.2.6	将输出 Tensor 转变为输出数组	18
4.2.7	找到最佳匹配的图片	18
4.3	WelcomeActivity	19
4.3.1	将 WelcomeActivity 声明为默认启动	19
4.3.2	自定义启动页面基本内容	19
4.3.3	使用 Handler 进行 Activity 切换	20
4.4	ToolBar and NavigationBar	21
4.4.1	自定义 ActionBar	21
4.4.2	将 xml 插件传给 MainActivity	22
4.4.3	丰富 toolbar	22
4.5	Camera	23
4.5.1	定制 CameraView	23

4.5.2 拍照后显示	24
4.5.3 recognizeImage 实现	24
4.6 Personal Web	25
4.6.1 制定 WebView	25
4.6.2 在 Info 以及个人简介中绑定 Button	26
4.6.3 创建 Web-View-Activity 接收内容	26
5 实验平台配置	27
5.1 Ubuntu 系统配置	27
5.1.1 Step1: Ubuntu LTS 16.04 ISO 下载	27
5.1.2 Step2: 虚拟机下安装 Ubuntu	27
5.1.3 Step3: 开启 vmware tools 服务	28
5.1.4 Step4: 配置 shadowsocks 客户端	29
5.2 Android Studio 与 Tensorflow 平台配置	31
5.2.1 Tensorflow 平台搭建	31
5.2.2 Android Studio 平台搭建	31
5.2.3 AS 与 TF 交互式平台搭建	32
6 项目测试	35
6.1 测试总体情况	35
6.2 基础显示测试	36
6.2.1 欢迎界面-主界面	36
6.2.2 个人简历-Personal Info	36
6.2.3 跳转界面-Github-Project-Weibo	36
6.3 相机运行测试	36
7 实验总结	39
7.1 基本介绍	39
7.2 能力提升	39
7.3 意识培养	39
7.4 认识不足	40
7.5 兴趣提升	40
7.6 注重想法	40
7.7 敢于开拓	40
8 参考文献	41
9 附录-源程序代码	45
9.1 build.gradle	45
9.2 AndroidManifest.xml	46
9.3 activity	47
9.3.1 MainActivity.java	47
9.3.2 WelcomeActivity.java	51
9.3.3 TensorFlowImageClassifier.java	53
9.3.4 InfoChinese.java	57
9.3.5 InfoEnglish.java	59

CONTENTS	5
----------	---

9.3.6	ToWebView.java	61
9.3.7	Classifier.java	62
9.4	layout	64
9.4.1	activity-main.xml	64
9.4.2	activity-welcome.xml	66
9.4.3	info-cn.xml	67
9.4.4	info-en.xml	69
9.4.5	nav-header.xml	72
9.4.6	web-view.xml	73
9.5	values	73
9.5.1	colors.xml	73
9.5.2	strings.xml	73
9.5.3	styles.xml	74
9.6	menu	74
9.6.1	nav-menu.xml	74
9.6.2	toolbar.xml	75

Chapter 1

关于本报告的说明

1.1 致谢

Give my sincere thanks to Teacher Du for his excellent teaching skills and serious altitude for our homework. Give my sincere thanks to some students who have helped us and who have inspired us when discussing with them. Give my sincere thanks to myself because I've overcome all difficulties and have successfully finished our embedded-software and android assignments – DeepCamera. Give my sincere thanks to those pioneers who have devoted themselves to writing immortal books. Give my sincere thanks to all the people sharing their ideas and harvests without pay in QA communities.

衷心感谢杜小智老师出色的教学风范以及严谨的治学态度，衷心感谢那些在讨论中帮助和启发我们的同学们，衷心感谢克服了种种困难最终完成了嵌入式软件与安卓设计综合实验的自己，衷心感谢那些写了不朽著作为后人指明道路的先驱们，衷心感谢那些在问答社区无私奉献自己智慧成果的所有同仁！

1.2 内容声明

本实验报告是西安交通大学 2015 级软件工程系本科生以《嵌入式软件设计与安卓综合实验》设计为契机进行的《Android-Machine Learning: DeepCamera》的设计与实现。所有工作均主要由本人完成。项目设计与实现过程中，严格采取工程模块化的设计思想，务必追求严谨，考虑周全。经过接近一个月的设计与实现，最终于 2017 年 6 月末完成整体设计，并完成所有相关代码以及样例运行。

关于该实验的相关程序以及发布在 GitHub 上，读者可以下载学习共同探讨心得，

项目地址为<https://github.com/WilliamYi96/DeepCamera>., 欢迎读者提出宝贵意见。

1.3 作者信息

易凯，西安交通大学软件工程专业软件 53 班。学号为 2151601053，联系邮箱 williamyi96@gmail.com. 主要研究兴趣为深度学习，机器视觉以及增强学习。现于智能网络与网络安全教育部重点实验室，西安交通大学软件学院创新实验室进行科研与工程训练。目前作为负责人完成国家级大学生创新创业训练课题。

座右铭：为天地立心，为生灵立命，为往圣继绝学，为万世开太平！

William Yi, who majors in software engineering in Xián Jiaotong University. He is interested in deep learning, computer vision and reinforcement learning. And he is the intent of Key Laboratory of Intelligent Network and Network Security, Ministry of Education in Xián Jiaotong University. The member of Innovative laboratory in software institute as well. Besides, He is conducting a national student project.

Motto: Pursue what you want and do want only you can do.

Chapter 2

实验研究基本背景

2.1 基本介绍

出于实际需要，实验研究背景部分进行简要说明。

由于 android 相对而言操作性更强，因此以下说明内容更多地放在 machine learning 之上。

从 2012 年来，以深度学习（其核心为卷积神经网络 CNN 和循环神经网络 RNN）为代表的机器学习引起了极大的轰动性。由于卷积神经网络与循环神经网络自身的特性，能够大大地提升各个领域的效率，其得到的结果相当惊艳，机器学习已经被应用于自然语言处理，机器视觉，语音识别，甚至是不对称信息博弈（代表为 AlphaGo 与德州扑克）。

TensorFlow 是 google 于 2015 年推出的机器学习平台，一经推出，即引起了很大反响，大大提升了机器学习实践的效率，如何该项目在 GitHub 上的 star 数已经超过了 6W，成为了机器学习框架中的佼佼者。

一直以来，目标识别（object detection）就是一个十分棘手的问题，没有任何或者少量先验知识进行物体识别的难度可想而知。而当前，为了有效地提升目标识别的准确度，学者们提出了各种各样的解决方案。

在这种大背景之下，萌生了将两者进行有机结合来设计制作 APP 的想法。

2.2 TensorFlow

TensorFlow 是一个采用数据流图（data flow graphs），用于数值计算的开源软件库。节点（Nodes）在图中表示数学操作，图中的线（edges）则表示在节点间相互联系的多维数据数组，即张量（tensor）。它灵活的架构让你可以在多种平台上展开计算，例如台式计算机中的一个或多个 CPU（或 GPU），服务器，移动设备等等。TensorFlow 最初由 Google 大脑小组（隶属于 Google 机器智能研究机构）的研究员和工程师们开发出来，用于机器学习和深度神经网络方面的研究，但这个系统的通用性使其也可广泛用于其他计算领域。

Tensorflow 有一个合理的 c++ 使用界面，也有一个易用的 python 使用界面来构建和执行我们的 graphs。我们可以直接写 python/c++ 程序，也可以用交互式的 ipython 界面来执行。

TensorFlow 基本情况可以概括为以下几点：

- TensorFlow 使用 Graph 来表示计算任务
- 在被称之为 Session 的 context 中执行图
- 使用 tensor 表示数据
- 使用 Variable 维护状态
- 使用 feed 和 fetch 可以为任意操作赋值或者从中获取数据

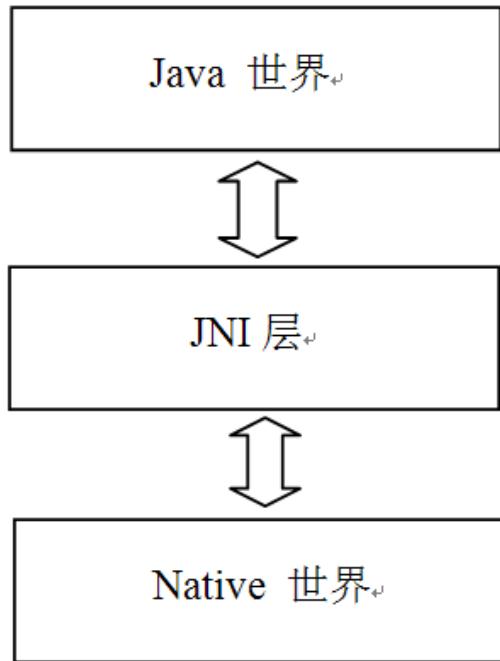
该机器学习框架以张量流的形式进行图运算，作为机器学习 library 中的重要成员，起到了极大作用。

2.3 JNI–DNK

JNI，是 Java Native Interface 的缩写，其是一种编程方式，中文为 Java 本地调用。通俗地说，JNI 是一种技术，通过这种技术可以做到以下两点：

- Java 程序中的函数可以调用 Native 语言写的函数，Native 一般指的是 C/C++ 编写的函数。
- Native 程序中的函数可以调用 Java 层的函数，也就是在 C/C++ 程序中可以调用 Java 的函数。

在 Android 平台上，JNI 就是一座将 Native 世界和 Java 世界间的天堑变成通途的桥，来看图 2-1，它展示了 Android 平台上 JNI 所处的位置：

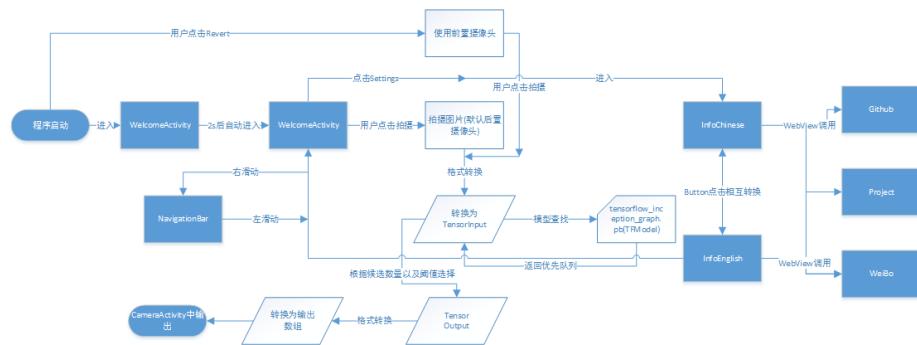


Android 的设计开发过程中继承了 JNI 这种编程方式，同时其发展为 NDK。NDK 全称是 Native Development Kit。NDK 提供了一系列的工具，帮助开发者快速开发 C（或 C++）的动态库，并能自动将 so 和 java 应用一起打包成 apk。这些工具对开发者的帮助是巨大的。

Chapter 3

项目总体设计

3.1 总体流程图



3.2 总体设计描述

在系统启动之后，首先进入 WelcomeActivity 状态，2s 之后跳转至 MainActivity。

在 MAinActivity 中，向右滑动可以进入 NavigationBar，然后向左滑动可以撤销上述操作。此外可以选择点击转换相机视角，从后置摄像头拍摄转换为前置摄像头拍摄。如果拍摄按键被执行，则进入模型处理、图片选择阶段。

图片选择阶段首先获取拍摄的照片，然后再将该照片转换为 TensorFlow 的 Input 格式，然后通过内置的模型以及数据标签进行分析，通过优先队列排出结果，然后将输出的结果 (Tensor Output Form) 转换为数组格式，通过添加的筛选个数以及阈值条件将结果进行输出。

WelcomeActivity 通过点击 settings 可直接进入到 InfoChinese 的 Activity，该 activity 中显示的是作者的基本信息。

InfoChinese 有回退键（直接回到 MAinActivity），也有切换键，直接进入到英文格式 (InfoEnglish)，还可以通过点击 button 访问作者的 Github page, project page 和 weibo page。

Chapter 4

项目模块化设计

4.1 TensorFlow 模型 Android 应用

我们要将下述内容中编译好了的 TensorFlow 文件.so 导入到 Android 应用中，然后通过 Android 的原生文件支持标准.jar 文件将.so 文件进行解析 (.jar 文件中存放的是调用 cpp 的 JAVA API)。

然后再将得到的 TensorFlow 的 Object Detection 对应的模型与对应的监督学习模式下的标签导入进来即可。

同时，注意上述导入过程中创建的项目要允许 C++ Support.

4.2 Image 分类识别实现

4.2.1 设置选择标准

我们仅仅选择通过 TensorFlow 模型得到的置信度最高的结果，其中最大选择数量为 3，同时设置置信阈值为 0.1f。如下所示：

```
1 private static final int MAX_RESULTS = 3;
2 private static final float THRESHOLD = 0.1f;
```

4.2.2 将物品标签写入内存

```
1 String actualFilename = labelFilename.split("file:///android_asset/")[1];
2 BufferedReader br = null;
3 br = new BufferedReader(new InputStreamReader(assetManager.open(actualFilename)));
4 String line;
5 while ((line = br.readLine()) != null) {
6     c.labels.add(line);
7 }
8 br.close();
```

recognizeImage 方法是实现图片识别的核心，采用的方式具体如下：

4.2.3 将图像数据转换为 bitMap 支持的数据格式

具体的转换关系如下：

```

1  bitmap.getPixels(intValues, 0, bitmap.getWidth(),
2                      0, 0, bitmap.getWidth(), bitmap.getHeight());
3  for (int i = 0; i < intValues.length; ++i) {
4      final int val = intValues[i];
5      floatValues[i * 3 + 0] = (((val >> 16) & 0xFF) - imageMean) / imageStd;
6      floatValues[i * 3 + 1] = (((val >> 8) & 0xFF) - imageMean) / imageStd;
7      floatValues[i * 3 + 2] = ((val & 0xFF) - imageMean) / imageStd;
8  }

```

4.2.4 将输入的数据导入 TensorFlow 模型

```

1  inferenceInterface.fillNodeFloat(
2      inputName, new int[]{1, inputSize, inputSize, 3}, floatValues);

```

4.2.5 运行模型推理得到输出 tensor

```
1  inferenceInterface.runInference(outputNames);
```

4.2.6 将输出 Tensor 转变为输出数组

```
1  inferenceInterface.readNodeFloat(outputName, outputs);
```

4.2.7 找到最佳匹配的图片

首先通过优先队列进行 TensorFlowFlow 库中图片置信度选择：

```

1  PriorityQueue<Recognition> pq =
2      new PriorityQueue<Recognition>(
3          3,
4          new Comparator<Recognition>() {
5              @Override
6              public int compare(Recognition lhs, Recognition rhs) {
7                  return Float.compare(rhs.getConfidence(), lhs.getConfidence());
8              }
9          });
10         .....

```

然后按照优先队列的顺序将大于阈值的图片输出筛选出来，如果没有，则输出为 unknown，若有则最多输出三个：

```

1 for (int i = 0; i < outputs.length; ++i) {
2     if (outputs[i] > THRESHOLD) {
3         pq.add(
4             new Recognition(
5                 "" + i, labels.size() > i ? labels.get(i) :
6                 "unknown", outputs[i], null));
7     }
8 }
```

通过如下方式进行数量比对：

```
1 int recognitionsSize = Math.min(pq.size(), MAX_RESULTS);
```

最后将结果调用 poll() 函数输出即可。

4.3 WelcomeActivity

欢迎模块实现的是应用程序启动时进入 WelcomeActivity 欢迎界面，经过 2s 延时之后进入 MainActivity 页面。其中可以通过在 WelcomeActivity 中使用 Thread 和 Handler 进行控制延时和启动。

4.3.1 将 WelcomeActivity 声明为默认启动

默认的启动方式是从 MainActivity 中启动，将其改为从 WelcomeActivity 中启动即可。修改的地方是 AndroidManifest.xml。

```

1 <activity android:name=".WelcomeActivity">
2     <intent-filter>
3         <action android:name="android.intent.action.MAIN" />
4         <category android:name="android.intent.category.LAUNCHER" />
5     </intent-filter>
6 </activity>
```

4.3.2 自定义启动页面基本内容

自定义启动主体是 LinearLayout，其中采用的启动背景画面是 welcome.png，启动时定义了两个 TextView，内容分别为 DeepCamera 和 Android with TensorFlow。

具体的参数如下：

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:tools="http://schemas.android.com/tools"
4     android:id="@+id/activity_welcome"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     android:background="@drawable/welcome"
8     android:orientation="vertical"
9     tools:context="com.deepcamera.williamyi.welcometest.WelcomeActivity">
```

```

10 <TextView
11     android:text="DeepCamera"
12     android:layout_width="match_parent"
13     android:layout_height="wrap_content"
14     android:textAlignment="center"
15     android:textColor="@android:color/holo_orange_light"
16     android:textSize="45dp"
17     android:layout_marginTop="100dp"
18     android:layout_alignParentTop="true"
19     android:layout_alignParentLeft="true"
20     android:layout_alignParentStart="true"
21     android:id="@+id/textView1" />
22 <TextView
23     android:text="Android with Tensorflow"
24     android:layout_width="match_parent"
25     android:layout_height="wrap_content"
26     android:textAlignment="center"
27     android:textColor="@android:color/holo_orange_light"
28     android:textSize="35dp"
29     android:layout_marginTop="10dp"
30     android:layout_alignParentTop="true"
31     android:layout_alignParentLeft="true"
32     android:layout_alignParentStart="true"
33     android:id="@+id/textView2" />
34 </LinearLayout>

```

4.3.3 使用 Handler 进行 Activity 切换

使用 Handler 进行 Activity 切换，并制定睡眠时间为 2s，同时使用到了 isShowWelcom()，便于之后进行是否显示 WelcomeActivity 的设置。

```

1 handler=new Handler(){
2     @Override
3     public void handleMessage(Message msg) {
4         Intent intent=new Intent(WelcomeActivity.this,MainActivity.class);
5         startActivity(intent);
6         WelcomeActivity.this.finish();
7     }
8 };
9
10 if (!isShowWelcome(this)) {
11     Intent intent=new Intent(this,MainActivity.class);
12     startActivity(intent);
13     this.finish();
14 } else{
15     new Thread(){
16         @Override
17         public void run() {

```

```

18     try {
19         Thread.sleep(2000);
20         handler.sendEmptyMessage(0);
21     } catch (InterruptedException e) {
22         e.printStackTrace();
23     }
24     }.start();
25 }
26 }
```

4.4 ToolBar and NavigationBar

ToolBar 和 NavigationBar 都是 MaterialDesign 中的内容，其联合使用可以大大增强安卓 UI 的整体效果。

4.4.1 自定义 ActionBar

由于任何新建的项目，默认都是会显示 ActionBar，但是 ActionBar 较为传统，灵活度不高。因此我们若想用 Material Design 中的内容，则需要在 style 中更改主题。

修改为如下所示：

```

1 <resources>
2
3 <!-- Base application theme. -->
4 <style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
5     <!-- Customize your theme here. -->
6     <item name="colorPrimary">@color/colorPrimary</item>
7     <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
8     <item name="colorAccent">@color/colorAccent</item>
9 </style>
10
11 </resources>
```

以上操作完成之后我们就成功地将 ActionBar 隐藏起来了，接下来修改 activity_main.java 中的代码，用 ToolBar 替代 ActionBar。

修改完成之后的内容如下所示：

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context="com.deepcamera.williamyi.welcometest.MainActivity">
8
9     <android.support.v7.widget.Toolbar
10        android:id="@+id/toolbar"
11        android:layout_width="match_parent"
```

```

12     android:layout_height="?attr/actionBarSize"
13     android:background="@color/colorPrimary"
14     android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
15     app:popupTheme="@style/ThemeOverlay.AppCompat.Light">
16   </android.support.v7.widget.Toolbar>
17
18 </FrameLayout>
```

4.4.2 将 xml 插件传给 MainActivity

通过 `findViewById()` 得到 Toolbar 的实例，然后调用 `setSupportActionBar()` 方法并将 Toolbar 实例传入，我们就可以使用 Toolbar 并且保持其外观上的一致。

4.4.3 丰富 toolbar

在 res 目录之下新建 menu，同时在文件夹下新建 toolbar.xml 文件，自定义 toolbar 中显示的内容。

如下所示：

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <menu xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto">
4   <item
5     android:id="@+id/backup"
6     android:icon="@drawable/ic_backup"
7     android:title="Backup"
8     app:showAsAction="always"/>
9   <item
10    android:id="@+id/delete"
11    android:icon="@drawable/ic_delete"
12    android:title="Delete"
13    app:showAsAction="ifRoom"/>
14   <item
15    android:id="@+id/settings"
16    android:icon="@drawable/ic_settings"
17    android:title="Settings"
18    app:showAsAction="never"/>
19 </menu>
```

接下来我们调用 `onCreateOptionsMenu()` 方法，并加载 toolbar.xml 菜单文件，然后在 `onOptionsItemSelected()` 方法中处理各个按钮的点击事件。

如下所示：

```

1 public boolean onCreateOptionsMenu(Menu menu) {
2     getMenuInflater().inflate(R.menu.toolbar, menu);
3     return true;
4 }
5
6 @Override
7 public boolean onOptionsItemSelected(MenuItem item) {
```

```

8   switch (item.getItemId()) {
9     case android.R.id.home:
10       mDrawerLayout.openDrawer(GravityCompat.START);
11       break;
12     case R.id.backup:
13       Toast.makeText(this, "You clicked Backup",
14                     Toast.LENGTH_SHORT).show();
15       break;
16     case R.id.delete:
17       Toast.makeText(this, "You clicked Delete",
18                     Toast.LENGTH_SHORT).show();
19       break;
20     case R.id.settings:
21       Toast.makeText(this, "You clicked Settings",
22                     Toast.LENGTH_SHORT).show();
23       break;
24     default:
25   }
26   return true;
27 }
```

关于 NavigationBar 由于 Android Studio 自身的特性，无需进行额外的设置，新建 activity 时直接选择 Navigation FrameLayout 即可。

4.5 Camera

4.5.1 定制 CameraView

Camera 的 UI 由四个部分构成，分别为 CameraView, ImageViewResult, ToggleButton 和 Object Detection Button。

其中 CameraView 采取的是一般化的布局，如下所示：

```

1 <com.flurgle.camerakit.CameraView
2   android:id="@+id/cameraView"
3   android:layout_width="300dp"
4   android:layout_height="300dp"
5   android:layout_marginBottom="32sp"
6   android:layout_marginTop="?attr actionBarSize"
7   android:layout_gravity="center|top" />
```

ImageViewResult 顾名思义就是用来显示拍摄的样例及其对应的结果的。其由一个 LinearLayout 作为大框架，同时该大框架由一个图片显示 ImageView 和一个文本匹配结果显示 TextView 构成，具体代码如下：

```

1 <LinearLayout
2   android:layout_marginLeft="8sp"
3   android:layout_width="match_parent"
4   android:layout_height="80dp"
5   android:layout_gravity="center|top"
6   android:layout_marginTop="380dp" />
```

```

7   android:gravity="center"
8   android:orientation="horizontal">
9
10  <ImageView
11    android:id="@+id/imageViewResult"
12    android:layout_width="75dp"
13    android:layout_height="75dp"
14    android:padding="2dp" />
15
16  <TextView
17    android:id="@+id/textViewResult"
18    android:layout_width="match_parent"
19    android:layout_height="80dp"
20    android:fadeScrollbars="false"
21    android:maxLines="15"
22    android:scrollbars="vertical"
23    android:gravity="center"
24    android:textColor="@android:color/black" />
25
26 </LinearLayout>
27
28      另外，两个Button较为简单，在此不进行赘述。
29 \end{small}

```

4.5.2 拍照后显示

```

1 cameraView.setCameraListener(new CameraListener() {
2     @Override
3     public void onPictureTaken(byte[] picture) {
4         super.onPictureTaken(picture);
5
6         Bitmap bitmap = BitmapFactory.decodeByteArray(picture, 0, picture.length);
7         bitmap = Bitmap.createScaledBitmap(bitmap, INPUT_SIZE, INPUT_SIZE, false);
8         imageViewResult.setImageBitmap(bitmap);
9         final List<Classifier.Recognition> results = classifier.recognizeImage(bitmap);
10        textViewResult.setText(results.toString());
11    }
12 });

```

使用 PictureTaken 函数得到的结果将其设置为 picture，然后通过 bitmap 控件，将其传递给 imageViewResult，同时使用 Classifier 这个类来实现图像识别（将图像传给该函数），将处理之后的结果传递给 textViewResult。

4.5.3 recognizeImage 实现

recognizeImage 方法是实现图片识别的核心，采用的方式具体如下：

- a. 将图像数据转换为 bitMap 支持的数据格式

具体的转换关系如下：

```

1 bitmap.getPixels(intValues, 0, bitmap.getWidth(),
2                     0, 0, bitmap.getWidth(), bitmap.getHeight());
3     for (int i = 0; i < intValues.length; ++i) {
4         final int val = intValues[i];
5         floatValues[i * 3 + 0] = (((val >> 16) & 0xFF) - imageMean) / imageStd;
6         floatValues[i * 3 + 1] = (((val >> 8) & 0xFF) - imageMean) / imageStd;
7         floatValues[i * 3 + 2] = ((val & 0xFF) - imageMean) / imageStd;
8     }

```

b. 将输入的数据导入 TensorFlow 模型

```

1 inferenceInterface.fillNodeFloat(
2     inputName, new int[]{1, inputSize, inputSize, 3}, floatValues);

```

c. 运行模型推理得到输出 tensor

```

1 inferenceInterface.runInference(outputNames);

```

d. 将输出 Tensor 转变为输出数组

```

1 inferenceInterface.readNodeFloat(outputName, outputs);

```

e. 找到最佳匹配的图片

首先通过优先队列进行 TensorFlow 库中图片置信度选择：

```

1 PriorityQueue<Recognition> pq =
2     new PriorityQueue<Recognition>(
3         3,
4         new Comparator<Recognition>() {
5             @Override
6             public int compare(Recognition lhs, Recognition rhs) {
7                 return Float.compare(rhs.getConfidence(), lhs.getConfidence());
8             }
9         });
10 .....

```

然后按照优先队列的顺序将大于阈值的图片输出筛选出来，如果没有，则输出为 unknown，若有则最多输出三个：

```

1 for (int i = 0; i < outputs.length; ++i) {
2     if (outputs[i] > THRESHOLD) {
3         pq.add(
4             new Recognition(
5                 "" + i, labels.size() > i ? labels.get(i) :
6                 "unknown", outputs[i], null));
7     }
8 }

```

通过如下方式进行数量比对：

```
1 int recognitionsSize = Math.min(pq.size(), MAX_RESULTS);
```

最后将结果调用 poll() 函数输出即可。

4.6 Personal Web

4.6.1 制定 WebView

主要是使用 WebView 控件，自定义如下：

```
1 <WebView
2     android:id="@+id/view_web"
3     android:layout_width="match_parent"
4     android:layout_height="match_parent">
5 </WebView>
```

4.6.2 在 Info 以及个人简介中绑定 Button

需要在个人简介中英文版中，当 button 受到点击时，进行 activity 的切换。由于中英两种实现方式相同，因此以下仅以中文版为例进行分析：

在 get 到 ID 之后，我们进行如下的 button 绑定：

```
1 mgithub.setOnClickListener(new View.OnClickListener() {
2     @Override
3     public void onClick(View v) {
4         Intent intent1 = new Intent();
5         intent1.setClass(InfoChinese.this, ToWebView.class);
6         intent1.putExtra("mwebsite", "mywebsite");
7         startActivity(intent1);
8     }
9});
```

其中 mywebsite 的内容分别为”<https://github.com/WilliamYi96>”，”<https://github.com/WilliamYi96/DeepCamera>”和”http://www.weibo.com/u/5794337545?from=page_100505_profile&wvr=6&mod=like&is_all=1”，其分别表示作者的 Github 主页，项目主页和微博主页。

4.6.3 创建 Web-View-Activity 接收内容

```
1 @Override
2 protected void onCreate(Bundle savedInstanceState) {
3     super.onCreate(savedInstanceState);
4     setContentView(R.layout.web_view);
5
6     Intent intent = this.getIntent();
7     Bundle bundle = intent.getExtras();
8     String value = bundle.getString("mwebsite");
```

```
9  
10    WebView webView = (WebView) findViewById(R.id.view_web);  
11    webView.getSettings().setJavaScriptEnabled(true);  
12    webView.setWebViewClient(new WebViewClient());  
13    webView.loadUrl(value);  
14 }
```

将该 Activity 与 web_view 进行绑定，同时使用 intent 接收来自点击 Button 之后来自 InfoActivity 中的内容，并且将这一内容作为网址传给新建的 webview，通过 WebView 控件进行显示。

Chapter 5

实验平台配置

5.1 Ubuntu 系统配置

以下内容为在 win10 以及 vmware workstations 12.0 的系统以及虚拟机支持下进行的 Ubuntu ets16.04 的安装。因此 win10 系统安装以及 vmware workstations 12.0 的安装在此省略，默认已经配置完成，直接进行 Ubuntu LTS 16.04 的安装与配置。

5.1.1 Step1: Ubuntu LTS 16.04 ISO 下载

从官网上，下载 Ubuntu ETS 16.04 的镜像文件。网址为 <https://www.ubuntu.com/download/desktop> (需要翻墙)

5.1.2 Step2: 虚拟机下安装 Ubuntu

不需要其他特别的配置，本人采取的是在移动硬盘中运行 Ubuntu。

相关配置如下：

▼ 设备	
内存	4.9 GB
处理器	4
硬盘(SCSI)	500 GB
CD/DVD (SATA)	正在使用文件 J:\...
网络适配器	NAT
USB 控制器	存在
声卡	自动检测
打印机	存在
显示器	自动检测

注意：网络连接采取桥接 (NAT) 的方式进行连接。

安装完成之后的环境如下所示：



5.1.3 Step3: 开启 vmware tools 服务

VMware tools 对于该项目的最大亮点是支持文件在虚拟机与物理机之间进行传输以及自动全屏扩展，因此其重要性不言而喻，之前使用过的 centos 系统，ubuntu gnome 16.04 都能够在虚拟机 —— 设置中点击 VMware tools 进行自动安装配置，而此处 vmware tools 由于是 ubuntu lts 16.04 或者其他缘故，因此需要手动安装。

手动安装的过程如下：

1. 使用 `ctrl + alt + T` 打开终端，再输入 `sudosu` 切换到 root 模式；
2. 将 ISO 挂载为 CD-ROM，(测试机的 ISO 在 `/media/williamyi/VMwareTools`)：
`mount /media/williamyi/VMware Tools /mnt/cdrom`
3. 用如下命令进行安装：

```
1 cd /tmp
```

```
2 tar -zxvf /mnt/cdrom/VMwareTools-.i386.tar.gz
```

4. 运行 VMware Tools 安装程序。

将目录更改到 vmware-tools-distrib, 然后运行./vmware-install.pl 即可。

5. 手动重启客户机 (本人测试时关闭了物理机更改才生效)

5.1.4 Step4: 配置 shadowsocks 客户端

shadowsocks 是啥, 在此就不再进行解释了。此部分相对而言较为复杂, 因为此前使用的代理均可以直接使用, 而 shadowsocks 不太人性化, 添加了许多额外的设置。其基本步骤如下:

a. 使用命令行安装 shadowsocks

```
1 apt-get install python-pip
2 pip install shadowsocks
```

b. 创建启动配置文件 shadowsocks.json

```
1 {
2   "server": "***.*.*.*.*",
3   "server_port":*****,
4   "local_address": "127.0.0.1",
5   "local_port":1080,
6   "password": "*****",
7   "timeout":300,
8   "method": "rc4-md5"
9 }
```

其中,

server 服务端的 IP

servier_port 服务端的端口

local_port 本地端口, 一般默认 1080

passwd ss 服务端设置的密码

timeout 超时设置和服务端一样

method 加密方法和服务端一样

c. 运行 shadowsocks.json

```
williamyi@williamyi-ml:~$ sslocal -c /home/williamyi/shadowsocks-libev/shadowsocks.json
INFO: loading config from /home/williamyi/shadowsocks-libev/shadowsocks.json
2017-06-26 20:15:10 INFO      loading libcrypto from libcrypto.so.1.0.0
2017-06-26 20:15:10 INFO      starting local at 127.0.0.1:1080
```

上述完成的是全局代理的配置, 以下为代理自动切换的配置:

d. 下载 SwitchyOmega 插件

建议使用的是 Chrome 浏览器，同时给 Chrome 浏览器安装 SwitchyOmega 插件，关于其插件我们也可以在 GitHub 上直接下载 (<https://github.com/FelisCatus/SwitchyOmega/releases/>)，然后浏览器地址打开 `chrome://extensions/`，将下载的插件进行安装即可。

e. 配置浏览器插件

新建代理，取名为 ss，配置如下：

Scheme	Protocol	Server	Port
(default)	SOCKS5	127.0.0.1	1080

Bypass List

Servers for which you do not want to use any proxy: (One server on each line.)
(Wildcards and more available...)

```
127.0.0.1
::1
localhost
```

ACTIONS

Apply changes
 Discard changes

然后点击到 auto switch，进行代理选择配置：

Profile :: auto switch

SETTINGS

Interface
 General
 Import/Export

PROFILES

proxy
 ss
 auto switch
 New profile...

ACTIONS

Apply changes

Rule list rules

(Any request matching the rule list below)

Rule List Format Switchy AutoProxy

Rule List URL https://raw.githubusercontent.com/gfwlist/gfwlist/master/gfwlist.txt

The rule list will be updated from this URL. If it is left blank, the following text will be parsed instead.

ACTIONS

Apply changes

其中使用的 rule list url 为：

<https://raw.githubusercontent.com/gfwlist/gfwlist/master/gfwlist.txt>

f. 代理运行

1 `sslocal -c *.shadowsocks.json`

至此大功告成。

5.2 Android Studio 与 Tensorflow 平台配置

5.2.1 Tensorflow 平台搭建

tensorflow 是谷歌公司开源的关于机器学习实践的平台，相关详细的介绍可以参考本人之前做的[总结](#)，以及 tensorflow[官方文档](#)。

以下使用 anaconda 进行 tensorflow 的安装：

a. 安装 anaconda

按[anaconda 官网](#)介绍进行安装。

b. 创建 tensorflow 的 anaconda 环境

```
1 $ conda create -n tensorflow
```

c. 激活 anaconda 环境

```
1 $ source activate tensorflow
2 (tensorflow)$ # The prompt should change
```

d. 用命令行在环境之下下载 tensorflow

```
1 (tensorflow)$ pip install --ignore-installed --upgrade \
2 https://storage.googleapis.com/tensorflow/linux/cpu/ \
3 tensorflow-1.2.0-cp36-cp36m-linux_x86_64.whl
```

此处下载的是 CPU only 的 python3.6 tensorflow 版本。

创建完成，此后可以在该环境之下使用

```
1 jupyter notebook
```

进行 ipython 的创作。

5.2.2 Android Studio 平台搭建

a. 下载并安装 android studio

在[官网](#)上下载 android studio。

安装是傻瓜式安装，较为简单，在此不进行赘述。

注：目前在 **Ubuntu16.04** 中，环境变量自动配置完成，因此无需进行额外操作（如果有变化依变化为准）

b. ubuntu 中启动 android studio

android studio 的启动路径为 `/android-studio/bin/`，然后直接在命令行中运行 `./studio.sh` 即可。

然后进行新工程的创建。

此处 android studio 可能卡在了 `rebuilding ****` 之上，导致这一问题的原因是 gradle 需要翻墙进行下载，方便的方式是我们先下载所需要的 gradle，然后再在创建工程的 `/gradle/wrapper/gradle-wrapper.properties` 中替换为如下文件内容：

```

1 #Mon Jun 26 00:35:23 CST 2017
2 distributionBase=GRADLE_USER_HOME
3 distributionPath=wrapper/dists
4 zipStoreBase=GRADLE_USER_HOME
5 zipStorePath=wrapper/dists
6 distributionUrl=gradle-3.3-all.zip

```

然后就可以很好的解决这个问题了。

我们创建的项目名为 DeepCamera，并进行勾选 C++ Support，android studio 会自动下载 ndk 的部分文件，完整版在[官网上](#)获取。

5.2.3 AS 与 TF 交互式平台搭建

a. 下载 tensorflow 源码

```

1 git clone --recurse-submodules \
2 https://github.com/tensorflow/tensorflow.git

```

b. WORKSPACE 文件修改

打开 tensorflow 文件夹下的 WORKSPACE，部分原文件为

```

1 # Uncomment and update the paths in these entries
2 # to build the Android demo.
3 android_sdk_repository(
4     name = "androidsdk",
5     api_level = 23,
6     build_tools_version = "25.0.1",
7     # Replace with path to Android SDK on your system
8     path = "<PATH_TO_SDK>",
9 )
10 #
11 android_ndk_repository(
12     name="androidndk",
13     path=<PATH_TO_NDK>,
14     api_level=14)

```

修改为：

```

1 # Uncomment and update the paths in these entries
2 # to build the Android demo.
3 android_sdk_repository(
4     name = "androidsdk",
5     api_level = 26,
6     # Ensure that you have the build_tools_version
7     # below installed in the
8     # SDK manager as it updates periodically.
9     build_tools_version = "26.0.0",
10    # Replace with path to Android SDK on your system

```

```

11   path = "/home/williamyi/Android/Sdk",
12 )
13
14 # Android NDK r12b is recommended
15     (higher may cause issues with Bazel)
16 android_ndk_repository(
17   name = "androidndk",
18   path = "/home/williamyi/Android/Ndk/android-ndk-r12b",
19   # This needs to be 14 or higher to compile TensorFlow.
20   # Note that the NDK version is not the API level.
21   api_level = 26)

```

c. 编译得到.so 文件

```

1  bazel build -c opt //tensorflow/contrib/android:libtensorflow_inference.so \
2      --crosstool_top=//external:android/crosstool \
3      --host_crosstool_top=@bazel_tools//tools/cpp:toolchain \
4      --cpu=armeabi-v7a

```

运行情况如下：

```

[576 / 850] Compiling tensorflow/core/ops/array_grad.cc
Slow read: a 849078502-byte read from /home/williamyi/.cache/bazel/_bazel_williamyi/c7fe9c0d242bea2f5e0d5f7952537aa4/execroot/tensorflow/bazel-out/arm-linux-androideabi-4.9-v7a-gnulibstdcpp-opt/bin/tensorflow/core/kernels/libandroid_tensorflow_kernels.lo took 9666ms.
Target //tensorflow/contrib/android:libtensorflow_inference.so up-to-date:
  bazel-bin/tensorflow/contrib/android/libtensorflow_inference.so
INFO: Elapsed time: 3427.648s, Critical Path: 2418.93s

```

该文件将在如下位置处找到：

```
1  bazel-bin/tensorflow/contrib/android/libtensorflow_inference.so
```

d. 编译得到.jar 文件

```
1  bazel-bin/tensorflow/contrib/android/libtensorflow_inference.so
```

运行情况如下：

```

INFO: Found 1 target...
Target //tensorflow/contrib/android:android_tensorflow_inference_java up-to-date:
  bazel-bin/tensorflow/contrib/android/libandroid_tensorflow_inference_java.jar
INFO: Elapsed time: 26.075s, Critical Path: 1.97s

```

该文件将在如下位置找到：

```
1  bazel-bin/tensorflow/contrib/android/libtensorflow_inference.so
```

注：在编译过程中报错的诸多问题发现都是 **WORKSPACE** 更改不恰当，比如 **ndk** 路径不正确，没有使用 **r12b**，使用版本不正确等，注意反复检查 **WORKSPACE**

Chapter 6

项目测试

6.1 测试总体情况

模块功能	情况	备注
欢迎界面		
Welcome 开机运行	成功	通过
Welcome 两秒后跳转至 MainActivity	成功	通过
主界面		
Camera 显示	成功	通过
切换相机	前后镜切换	通过
探测物体	拍摄照片并显示	通过
点击 Backup 后 Toast Backup	Toast Backup	通过
点击 Delete 后 Toast Delete	Toast Delete	通过
点击 Settings 后 Toast Settings	Toast Settings	通过
个人简历		
简历界面显示	成功	通过
Back 键	返回到 Camera	通过
English 键	点击转换为英文	通过
Github 键	在当前应用打开个人 GitHub	通过
Project 键	打开本项目网站	通过
Weibo 键	打开个人微博	通过
Personal Info		
简历英文显示	成功	通过
Back 键	返回到 Camera	通过
中文键	点击转换为中文	通过
Github 键	在当前应用打开个人 GitHub	通过
Project 键	打开本项目网站	通过
Weibo 键	打开个人微博	通过
通过率	100%	

Table 6.1: 测试总体情况

6.2 基础显示测试

6.2.1 欢迎界面-主界面

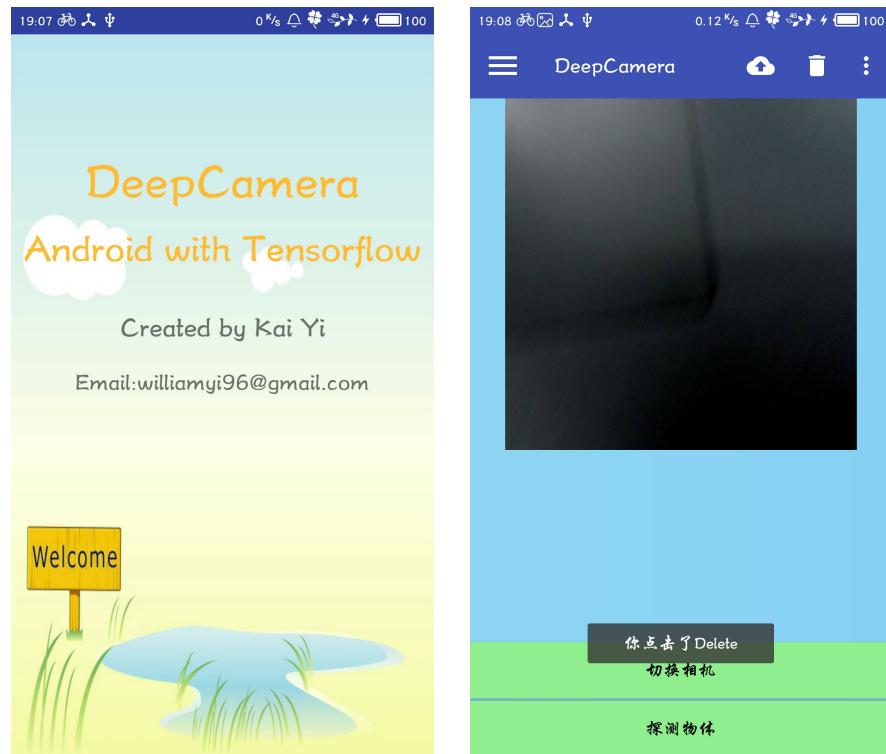


Figure 6.1: 欢迎界面-主界面

6.2.2 个人简历-Personal Info

6.2.3 跳转界面-Github-Project-Weibo

6.3 相机运行测试

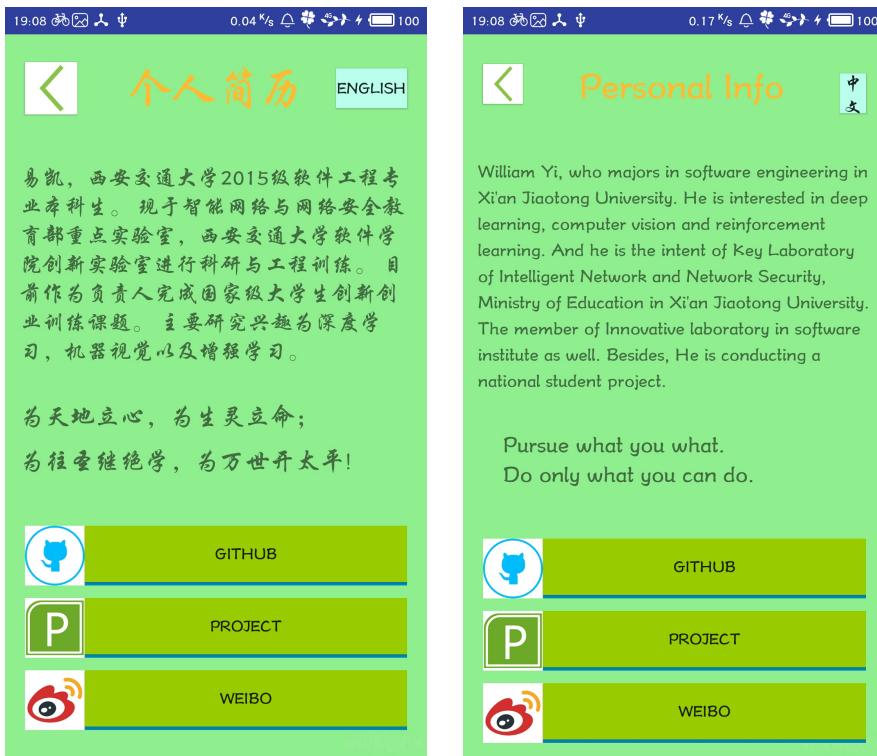


Figure 6.2: 欢迎界面-主界面

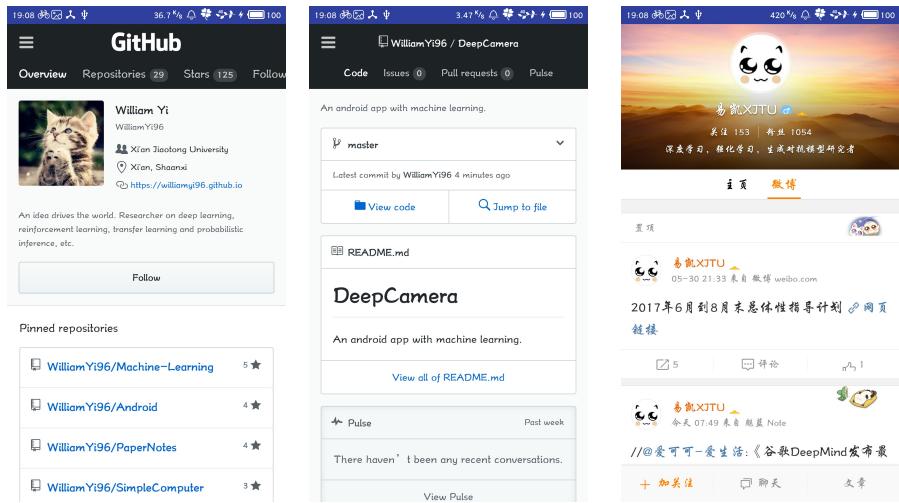


Figure 6.3: 跳转界面

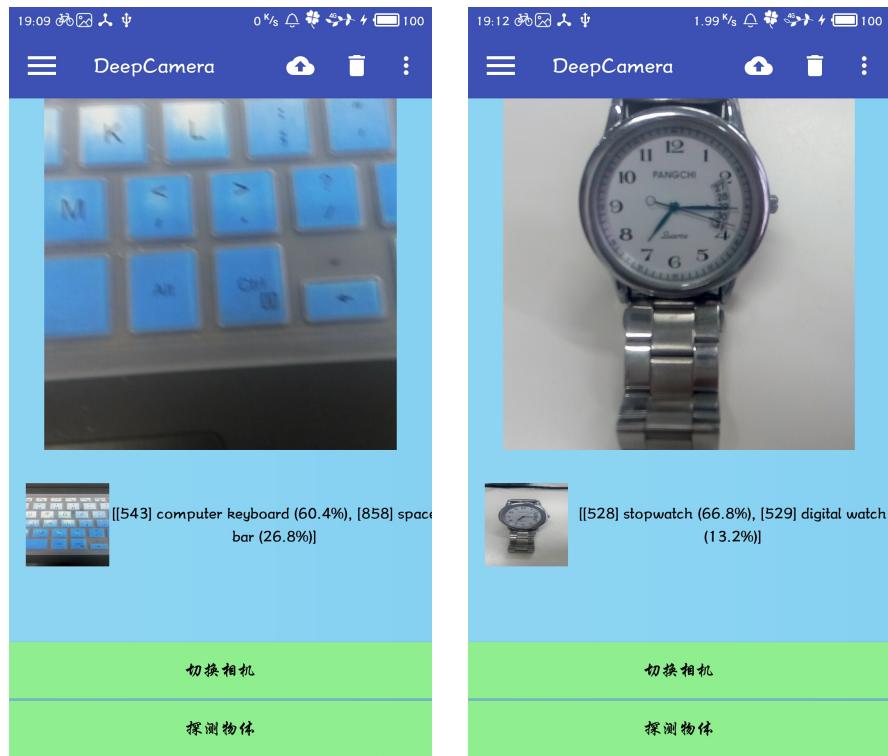


Figure 6.4: 相机运行测试结果 1

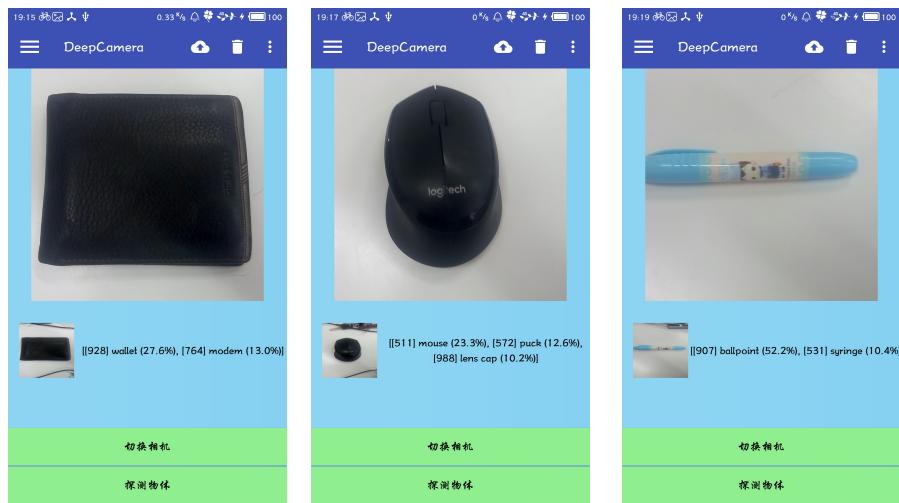


Figure 6.5: 相机运行测试结果 2

Chapter 7

实验总结

7.1 基本介绍

为时一个学期的《嵌入式软件设计》课程的学习就这样结束了，这一路走来，受益良多。通过嵌入式软件以及安卓应用开发的学习，对于相关知识有了较为深入的理解。在大作业是选择嵌入式软件设计还是安卓应用开发的节骨眼上，起初我选择了嵌入式软件设计，原本想借此机会对于计算机有一个深入的理解，慢慢发现底层的东西理解难度是一方面，另外一方面，遇到了有些错误竟然也不知道如何去解决，后来忍痛选择了安卓应用开发。但是为了将两者贴合起来，因此我决定初次使用 ndk 进行程序设计。

由于本人目标是成为一名出色的机器学习科学家，因此想要尝试在保留安卓基础程序设计的框架基础上，将机器学习应用于安卓程序设计来写一个程序，由此来促进对于机器学习以及 android 的认识。这一路走来，从零开始，装系统，配环境，修 bug，敲代码，搞创造，遇到了诸多问题，同时自己也受益良多。

我将从能力提升，意识培养，认识不足，兴趣提升，注重想法，敢于开拓这六个角度进行归纳总结：

7.2 能力提升

虽然我们是软件工程专业的本科生，但是嵌入式软件设计的学习过程中我们掌握了很多软硬件接口设计的基本方法，同时主要对安卓 ndk 开发的过程中，对于计算机，对于安卓，对于机器学习有了更为深入的认识。一方面像装系统，配环境，通过看日志解决 bug 等技能得到了进一步的强化。

7.3 意识培养

杜老师以及毛老师上课反复强调，嵌入式设计，安卓应用开发是一种思维方法，是一种意识，是大家的按照工程设计的基本流程进行分析问题，解决问题的意识，这种意识才是比知识更为重要的东西。老师在课堂之上循循善诱，逐步地引导我们对计算机、对软硬件接口设计，对嵌入式，对安卓应用程序设计进行

深入理解，同时在课堂上经常抛出许多富有创造性又富有挑战性的问题，在这个过程中，培养了自己静下心来解决问题的能力。

7.4 认识不足

在嵌入式软件设计课程的学习以及安卓应用程序设计的过程中，这段经历总体而言是非常有趣的，很多的老师精选的例题不仅具有代表性，同时不乏趣味性，让我打开眼界，为之后的进一步学习，以及将来的嵌入式软件设计与安卓应用开发的进一步学习奠定了坚实的基础。

在完成实验报告的过程中，总体思路虽然没有发生改变，但是细节上的实现遇到了诸多的问题，而且在实际地单独对于某些问题进行系统分析的过程之中，发现实际上遇到了一些阻碍，同时也说明了自己的能力有待提升。虽然这是初次进行 androidndk 的开发，总体而言该应用程序相对基础，相对简单，而相对而言实际的 ndk，机器学习的安卓应用开发，更具有挑战性，之后要加强这方面能力的提升以及强化训练。

7.5 兴趣提升

嵌入式软件设计以及安卓应用开发个人认为更多的是在兴趣驱动之下进行分析问题解决问题的过程，在这个过程中，遇到了很多新奇有趣的小技巧，同时也运用了许多让人眼前一亮的方法，极大地提升了自己进一步学习嵌入式以及安卓程序设计，并提升分析问题，系统总体设计能力的兴趣。

7.6 注重想法

安卓应用开发的核心在于进行模块化的设计，同时在知晓全局的基础之上进行模块的实现，然后再将各个模块进行集成。往往好的模块规范决定了设计的成败与否，因此需要敢于去提出自己的想法，然后不断地修正，以期臻于完美。

同时，由于 android 当下具有相当多的框架，因此在 idea 的驱动之下，如何将他们用得恰到好处是一门值得分外注意的学问，需要不断地去提升能力。同时，想法应该成为安卓应用开发的核心。

7.7 敢于开拓

安卓程序设计，或者说嵌入式软件设计的方法不是唯一的，或许能够得到满意的结果的都是合理化的方法，但是在这个过程中，如何加强工程化思想的渗透，同时设计得成一门艺术，则需要不断地去修正，不断地去开拓创新，这样才能够不断地完善。

同时，其他课程以及科研训练的真谛也在于此，真正的魅力不在于我马上就可以找到解决问题的最好方法，而是在不断地摸索中，提升了个人的能力，培养了开拓创新的精神，这才是社会赖以生存发展的本质意义。

Chapter 8

参考文献

1. Ubuntu ETS 16.04 镜像文件下载地址:<https://www.ubuntu.com/download/desktop>
2. 从 ISO 映像手动安装 VMware Tools (2075647). https://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=2075647
3. 安装和配置 VMware Tools. https://www.vmware.com/files/cn/pdf/vmware-tools-installation-configuration_CN.pdf
4. 奇影加速器. <http://www.qying.net/>
5. linux-ubuntu 使用 shadowsocks 客户端配置. <http://www.qying.net/cms/33>
6. Guide to Android TensorFlow Machine Learning Example. <https://blog.mindorks.com/android-tensorflow-machine-learning-example-ff0e9b2654cc>
7. TensorFlow. <https://www.tensorflow.org/>
8. MNIST_for_ML_beginners\$Experts.md. https://github.com/WilliamYi96/Machine-Learning/blob/master/ML_Theories/Tensorflow/MNIST_for_ML_beginners%24Experts.md
9. Installing TensorFlow on Ubuntu. https://www.tensorflow.org/install/install_linux#installing_with_anaconda
10. DOWNLOAD ANACONDA. <https://www.continuum.io/downloads>
11. Download Android Studio. <https://developer.android.com/studio/index.html>

12. Guide to Android Studio. <https://developer.android.com/studio/intro/index.html>
13. NDK Download. <https://developer.android.com/ndk/downloads/index.html>
14. Guide to NDK. <https://developer.android.com/ndk/guides/index.html>
15. Create Android with C/C++. <https://developer.android.com/studio/projects/add-native-code.html>
16. Bazel. <https://bazel.build/versions/master/docs/install.html>
17. Latex 中文字体. http://blog.sina.com.cn/s/blog_6d01f5790100rui9.html
18. MultiModel: Multi-Task Machine Learning Across Domains. Google Research.
<https://research.googleblog.com/2017/06/multimodel-multi-task-machine-learning.html>
19. Supercharge your Computer Vision models with the TensorFlow Object Detection API.
<https://research.googleblog.com/2017/06/supercharge-your-computer-vision-models.html>
20. YouTube-BoundingBoxes: A Large High-Precision Human-Annotated Data Set for Object Detection in Video, Real et al., CVPR 2017.
21. Speed/accuracy trade-offs for modern convolutional object detectors, Huang et al., CVPR 2017.
22. MobileNets: Efficient convolutional neural networks for mobile vision applications, Howard et al., arXiv preprint arXiv:1704.04861, 2017.
23. MobileNets: Open-Source Models for Efficient On-Device Vision. <https://research.googleblog.com/2017/06/mobilenets-open-source-models-for.html>
24. Tensorflow Github. <https://github.com/tensorflow/tensorflow>
25. [Android Studio 权威教程] 断点调试和高级调试. <http://blog.csdn.net/yy1300326388/article/details/46501871>

26. JNI 开发流程. <http://wiki.jikexueyuan.com/project/jni-ndk-developer-guide/workflow.html>
27. How to include *.so library in Android Studio?. <https://stackoverflow.com/questions/24357687/how-to-include-so-library-in-android-studio>
28. The following classes could not be instantiated: - android.support.v7.widget.Toolbar.
<https://stackoverflow.com/questions/26575815/the-following-classes-could-not-be-instantiated-android-support-v7-widget-too>
29. Cannot replace default icon in android studio. StackOverflow. <https://stackoverflow.com/questions/30563900/cannot-replace-default-icon-in-android-studio>
30. Android TensorFlow Machine Learning Example. <https://github.com/MindorksOpenSource/AndroidTensorFlowMachineLearningExample>
31. 郭霖. <第一行代码>(第二版). 人民邮电出版社. 2016-12.
32. From Java to Kotlin. <https://github.com/MindorksOpenSource/from-java-to-kotlin>
33. 多个 Activity 之间切换原理介绍. <http://blog.csdn.net/xiazdong/article/details/7749261>
34. N. Dalal; B. Triggs. Histograms of oriented gradients for human detection.
35. Zheng Song; Qiang Chen; Zhongyang Huang; Yang Hua; Shuicheng Yan. Contextualizing object detection and classification.
36. Seyed Eghbal Ghobadi; Klaus Hartmann; Wolfgang Weihs; Chayakorn Netramai; Otmar Loffeld; Hubert Roth. Detection and Classification of Moving Objects-Stereo or Time-of-Flight Images.
37. Mastering the Game of Go with Deep Neural Networks and Tree Search <https://gogameguru.com/i/2016/03/deepmind-mastering-go.pdf>
38. Rosin, C. D. Multi-armed bandits with episode context. Annals of Mathematics and Artificial Intelligence 61, 203–230 (2011).
39. Lanctot, M., Winands, M. H. M., Pepels, T. & Sturtevant, N. R. Monte Carlo tree search with heuristic evaluations using implicit minimax backups. In IEEE Conference on Computational Intelligence and Games, 1–8 (2014).

40. Huang, S. & Muller, M. Investigating the limits of Monte-Carlo tree search methods in computer Go. In 8th International Conference on Computers and Games, 39–48 (2013).
41. Big Picture Machine Learning: Classifying Text with Neural Networks and TensorFlow.
<https://medium.freecodecamp.org/big-picture-machine-learning-classifying-text-with-neural-networks-and-tensorflow-d94036ac2274>
42. TensorFlow Android Camera Demo. <https://github.com/tensorflow/tensorflow/tree/799e31f3840c21322e380e1ec6e5bacb95d016fa/tensorflow/examples/android>
43. JNI Tips. <https://developer.android.com/training/articles/perf-jni.html>
44. An Introduction to Practical JNI. <http://wiki.jikexueyuan.com/project/deep-android-v1/jni.html>

Chapter 9

附录—源程序代码

9.1 build.gradle

```
1 apply plugin: 'com.android.application'
2
3 android {
4     compileSdkVersion 25
5     buildToolsVersion "26.0.0"
6     defaultConfig {
7         applicationId "com.deepcamera.williamyi.deepcamera"
8         minSdkVersion 19
9         targetSdkVersion 25
10        versionCode 1
11        versionName "1.0"
12        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
13        externalNativeBuild {
14            cmake {
15                cppFlags ""
16            }
17        }
18    }
19    buildTypes {
20        release {
21            minifyEnabled false
22            proguardFiles getDefaultProguardFile('proguard-android.txt'),
23            'proguard-rules.pro'
24        }
25    }
26    externalNativeBuild {
27        cmake {
28            path "CMakeLists.txt"
29        }
30    }
31 }
```

```

32
33 dependencies {
34     compile fileTree(dir: 'libs', include: ['*.jar'])
35     androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
36         exclude group: 'com.android.support', module: 'support-annotations'
37     })
38     compile files('libs/libandroid_tensorflow_inference_java.jar')
39     compile 'com.android.support:appcompat-v7:25.3.1'
40     compile 'com.android.support.constraint:constraint-layout:1.0.2'
41     compile 'org.litepal.android:core:1.4.1'
42     compile 'com.squareup.okhttp3:okhttp:3.4.1'
43     compile 'com.google.code.gson:gson:2.7'
44     compile 'com.github.bumptech.glide:glide:3.7.0'
45     compile 'com.flurgle:camerakit:0.9.13'
46     compile 'junit:junit:4.12'
47     compile 'com.android.support:design:25.3.1'
48     compile 'de.hdodenhof:circleimageview:2.1.0'
49     compile 'com.android.support:support-v4:25.3.1'
50     compile 'com.android.support:support-vector-drawable:25.3.1'
51     testCompile 'junit:junit:4.12'
52 }
```

9.2 AndroidManifest.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3     package="com.deepcamera.williamyi.deepcamera">
4
5     <uses-permission android:name="android.permission.INTERNET">
6         </uses-permission>
7     <uses-permission android:name="android.permission.ACCESS_WIFI_STATE">
8         </uses-permission>
9     <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE">
10        </uses-permission>
11     <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW">
12         </uses-permission>
13     <uses-permission android:name="android.permission.WRITE_SYNC_SETTINGS">
14         </uses-permission>
15     <uses-permission android:name="android.permission.READ_SYNC_SETTINGS">
16         </uses-permission>
17
18     <application
19         android:allowBackup="true"
20         android:icon="@mipmap/ic_launcher"
21         android:label="@string/app_name"
22         android:roundIcon="@mipmap/ic_launcher_round"
23         android:supportsRtl="true"
24         android:theme="@style/AppTheme">
```

```
25 <activity android:name=".WelcomeActivity">
26     <intent-filter>
27         <action android:name="android.intent.action.MAIN" />
28
29         <category android:name="android.intent.category.LAUNCHER" />
30     </intent-filter>
31 </activity>
32 <activity android:name=".MainActivity" ></activity>
33 <activity android:name=".InfoChinese" ></activity>
34 <activity android:name=".InfoEnglish" ></activity>
35 <activity android:name=".ToWebView" ></activity>
36 </application>
37
38 </manifest>
```

9.3 activity

9.3.1 MainActivity.java

```
1 package com.deepcamera.williamyi.deepcamera;
2
3 /**
4 * Created by williamyi on 2017/6/18.
5 * Email: williamyi96@gmail.com
6 */
7
8 import android.content.Intent;
9 import android.graphics.Bitmap;
10 import android.graphics.BitmapFactory;
11 import android.os.Bundle;
12 import android.support.annotation.NonNull;
13 import android.support.design.widget.NavigationView;
14 import android.support.v4.view.GravityCompat;
15 import android.support.v4.widget.DrawerLayout;
16 import android.support.v7.app.ActionBar;
17 import android.support.v7.app.AppCompatActivity;
18 import android.support.v7.widget.Toolbar;
19 import android.text.method.ScrollingMovementMethod;
20 import android.view.Menu;
21 import android.view.MenuItem;
22 import android.view.View;
23 import android.widget.Button;
24 import android.widget.ImageView;
25 import android.widget.TextView;
26 import android.widget.Toast;
27
28 import com.flurgle.camerakit.CameraListener;
29 import com.flurgle.camerakit.CameraView;
```

```
30  
31 import java.util.List;  
32 import java.util.concurrent.Executor;  
33 import java.util.concurrent.Executors;  
34  
35 public class MainActivity extends AppCompatActivity {  
36  
37     private static final int INPUT_SIZE = 224;  
38     private static final int IMAGE_MEAN = 117;  
39     private static final float IMAGE_STD = 1;  
40     private static final String INPUT_NAME = "input";  
41     private static final String OUTPUT_NAME = "output";  
42  
43     private static final String MODEL_FILE =  
44         "file:///android_asset/tensorflow_inception_graph.pb";  
45     private static final String LABEL_FILE =  
46         "file:///android_asset/imagenet_comp_graph_label_strings.txt";  
47  
48     private Classifier classifier;  
49     private Executor executor = Executors.newSingleThreadExecutor();  
50     private TextView textViewResult;  
51     private Button btnDetectObject, btnToggleCamera;  
52     private ImageView imageViewResult;  
53     private CameraView cameraView;  
54  
55     private Button backCamera;  
56     private Button toChinese;  
57     private Button toEnglish;  
58  
59     private DrawerLayout mDrawerLayout;  
60  
61     @Override  
62     protected void onCreate(Bundle savedInstanceState) {  
63         super.onCreate(savedInstanceState);  
64         setContentView(R.layout.activity_main);  
65         cameraView = (CameraView) findViewById(R.id.cameraView);  
66         imageViewResult = (ImageView) findViewById(R.id.imageViewResult);  
67         textViewResult = (TextView) findViewById(R.id.textViewResult);  
68         textViewResult.setMovementMethod(new ScrollingMovementMethod());  
69  
70         btnToggleCamera = (Button) findViewById(R.id.btnToggleCamera);  
71         btnDetectObject = (Button) findViewById(R.id.btnDetectObject);  
72         backCamera = (Button) findViewById(R.id.back_camera);  
73         toEnglish = (Button) findViewById(R.id.change_to_English);  
74         toChinese = (Button) findViewById(R.id.change_to_Chinese);  
75  
76         Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);  
77         setSupportActionBar(toolbar);  
78         mDrawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);  
79         NavigationView navView = (NavigationView) findViewById(R.id.nav_view);
```

```
80    ActionBar actionBar = getSupportActionBar();
81    if(actionBar != null) {
82        actionBar.setDisplayHomeAsUpEnabled(true);
83        actionBar.setHomeAsUpIndicator(R.drawable.ic_menu);
84    }
85    navView.setCheckedItem(R.id.nav_call);
86    navView.setNavigationItemSelectedListener(
87        new NavigationView.OnNavigationItemSelected() {
88            @Override
89            public boolean onNavigationItemSelected(@NonNull MenuItem item) {
90                mDrawerLayout.closeDrawers();
91                return true;
92            }
93        });
94
95    cameraView.setCameraListener(new CameraListener() {
96        @Override
97        public void onPictureTaken(byte[] picture) {
98            super.onPictureTaken(picture);
99
100            Bitmap bitmap = BitmapFactory.decodeByteArray(picture, 0, picture.length);
101
102            bitmap = Bitmap.createScaledBitmap(bitmap, INPUT_SIZE, INPUT_SIZE, false);
103
104            imageViewResult.setImageBitmap(bitmap);
105
106            final List<Classifier.Recognition> results =
107                classifier.recognizeImage(bitmap);
108
109            textViewResult.setText(results.toString());
110        }
111    });
112
113    btnToggleCamera.setOnClickListener(new View.OnClickListener() {
114        @Override
115        public void onClick(View v) {
116            cameraView.toggleFacing();
117        }
118    });
119
120    btnDetectObject.setOnClickListener(new View.OnClickListener() {
121        @Override
122        public void onClick(View v) {
123            cameraView.captureImage();
124        }
125    });
126
127    initTensorFlowAndLoadModel();
128
129 }
```

```
130     @Override
131     protected void onResume() {
132         super.onResume();
133         cameraView.start();
134     }
135
136     @Override
137     protected void onPause() {
138         cameraView.stop();
139         super.onPause();
140     }
141
142     @Override
143     protected void onDestroy() {
144         super.onDestroy();
145         executor.execute(new Runnable() {
146             @Override
147             public void run() {
148                 classifier.close();
149             }
150         });
151     }
152
153     private void initTensorFlowAndLoadModel() {
154         executor.execute(new Runnable() {
155             @Override
156             public void run() {
157                 try {
158                     classifier = TensorFlowImageClassifier.create(
159                         getAssets(),
160                         MODEL_FILE,
161                         LABEL_FILE,
162                         INPUT_SIZE,
163                         IMAGE_MEAN,
164                         IMAGE_STD,
165                         INPUT_NAME,
166                         OUTPUT_NAME);
167                     makeButtonVisible();
168                 } catch (final Exception e) {
169                     throw new RuntimeException(
170                         "Error initializing TensorFlow!", e);
171                 }
172             }
173         });
174     }
175
176     private void makeButtonVisible() {
177         runOnUiThread(new Runnable() {
178             @Override
179             public void run() {
```

```

180         btnDetectObject.setVisibility(View.VISIBLE);
181     }
182   });
183 }
184
185 public boolean onCreateOptionsMenu(Menu menu) {
186   getMenuInflater().inflate(R.menu.toolbar, menu);
187   return true;
188 }
189
190 @Override
191 public boolean onOptionsItemSelected(MenuItem item) {
192   switch (item.getItemId()) {
193     case R.id.home:
194       Toast.makeText(this, "你点击了 Home",
195                   Toast.LENGTH_SHORT).show();
196       mDrawerLayout.openDrawer(GravityCompat.START);
197       Intent intent = new Intent();
198 //       intent.setClass(MainActivity.this, MainActivity.class);
199       intent.putExtra("name", "williamyi");
200       startActivity(intent);
201       break;
202     case R.id.backup:
203       Toast.makeText(this,
204                     "你点击了 Backup", Toast.LENGTH_SHORT).show();
205       break;
206     case R.id.delete:
207       Toast.makeText(this,
208                     "你点击了 Delete", Toast.LENGTH_SHORT).show();
209       break;
210     case R.id.information:
211       Toast.makeText(this,
212                     "你点击了 Information", Toast.LENGTH_SHORT).show();
213       Intent intent1 = new Intent();
214       intent1.setClass(MainActivity.this, InfoChinese.class);
215       intent1.putExtra("name", "williamyi");
216       startActivity(intent1);
217       break;
218     default:
219   }
220   return true;
221 }
222 }
```

9.3.2 WelcomeActivity.java

```

1 package com.deepcamera.williamyi.deepcamera;
2
3 /**
```

```
4  * Created by williamyi on 2017/6/14.
5  * Email: williamyi96@gmail.com
6  */
7
8 import android.content.Context;
9 import android.content.Intent;
10 import android.content.SharedPreferences;
11 import android.os.Bundle;
12 import android.os.Handler;
13 import android.os.Message;
14 import android.support.v7.app.AppCompatActivity;
15
16 import android.content.Context;
17 import android.content.Intent;
18 import android.content.SharedPreferences;
19 import android.os.Handler;
20 import android.os.Message;
21 import android.support.v7.app.AppCompatActivity;
22 import android.os.Bundle;
23
24 public class WelcomeActivity extends AppCompatActivity {
25     Handler handler;
26     @Override
27     protected void onCreate(Bundle savedInstanceState) {
28         super.onCreate(savedInstanceState);
29         setContentView(R.layout.activity_welcome);
30
31         handler=new Handler(){
32             @Override
33             public void handleMessage(Message msg) {
34                 Intent intent=
35                     new Intent(WelcomeActivity.this,MainActivity.class);
36                 startActivity(intent);
37                 WelcomeActivity.this.finish();
38             }
39         };
40
41         if (!isShowWelcome(this)) {
42             Intent intent=new Intent(this,MainActivity.class);
43             startActivity(intent);
44             this.finish();
45         }
46     else{
47         new Thread(){
48             @Override
49             public void run() {
50                 try {
51                     Thread.sleep(2000);
52                     handler.sendEmptyMessage(0);
53                 } catch (InterruptedException e) {
```

```

54             e.printStackTrace();
55         }
56     }
57     }.start();
58 }
59
60 }
61
62
63     public static final String APP_NAME="myapplication";
64     public static final String KEY_SHOW_WELCOME="show_welcome";
65     public static boolean isShowWelcome(Context context){
66         //1.GET SP
67         SharedPreferences sp=
68             context.getSharedPreferences(APP_NAME,MODE_PRIVATE);
69         //2.GET DATA BY AND SET THE DEFAULT VALUE
70         return sp.getBoolean(KEY_SHOW_WELCOME,true);
71     }
72
73 }
```

9.3.3 TensorFlowImageClassifier.java

```

1 package com.deepcamera.williamyi.deepcamera;
2
3 /**
4 * Created by williamyi on 2017/6/22.
5 * Email: williamyi96@gmail.com
6 */
7
8
9 import android.content.res.AssetManager;
10 import android.graphics.Bitmap;
11 import android.os.Trace;
12 import android.util.Log;
13
14 import org.tensorflow.contrib.android.TensorFlowInferenceInterface;
15
16 import java.io.BufferedReader;
17 import java.io.IOException;
18 import java.io.InputStreamReader;
19 import java.util.ArrayList;
20 import java.util.Comparator;
21 import java.util.List;
22 import java.util.PriorityQueue;
23 import java.util.Vector;
24
25 /**
26 * A classifier specialized to label images using TensorFlow.
```

```
27  /*
28  public class TensorFlowImageClassifier implements Classifier {
29
30     private static final String TAG = "TensorFlowImageClassifier";
31
32     // Only return this many results with at least this confidence.
33     private static final int MAX_RESULTS = 3;
34     private static final float THRESHOLD = 0.1f;
35
36     // Config values.
37     private String inputName;
38     private String outputName;
39     private int inputSize;
40     private int imageMean;
41     private float imageStd;
42
43     // Pre-allocated buffers.
44     private Vector<String> labels = new Vector<String>();
45     private int[] intValues;
46     private float[] floatValues;
47     private float[] outputs;
48     private String[] outputNames;
49
50     private TensorFlowInferenceInterface inferenceInterface;
51
52     private TensorFlowImageClassifier() {
53     }
54
55     /**
56      * Initializes a native TensorFlow session for classifying images.
57      *
58      * @param assetManager
59      *        The asset manager to be used to load assets.
60      * @param modelName
61      *        The filepath of the model GraphDef protocol buffer.
62      * @param labelFilename
63      *        The filepath of label file for classes.
64      * @param inputSize
65      *        The input size. A square image of inputSize x inputSize is assumed.
66      * @param imageMean
67      *        The assumed mean of the image values.
68      * @param imageStd
69      *        The assumed std of the image values.
70      * @param inputName
71      *        The label of the image input node.
72      * @param outputName
73      *        The label of the output node.
74      * @throws IOException
75      */
76     public static Classifier create(
```

```

77     AssetManager assetManager,
78     String modelFilename,
79     String labelFilename,
80     int inputSize,
81     int imageMean,
82     float imageStd,
83     String inputName,
84     String outputName)
85     throws IOException {
86     TensorFlowImageClassifier c = new TensorFlowImageClassifier();
87     c.inputName = inputName;
88     c.outputName = outputName;
89
90     // Read the label names into memory.
91     // TODO(andrewsharp): make this handle non-assets.
92     String actualFilename = labelFilename.split(
93             "file:///android_asset/")[1];
94     Log.i(TAG, "Reading labels from: " + actualFilename);
95     BufferedReader br = null;
96     br = new BufferedReader(
97             new InputStreamReader(assetManager.open(actualFilename)));
98     String line;
99     while ((line = br.readLine()) != null) {
100         c.labels.add(line);
101     }
102     br.close();
103
104     c.inferenceInterface = new TensorFlowInferenceInterface();
105     if (c.inferenceInterface.initializeTensorFlow(
106             assetManager, modelFilename) != 0) {
107         throw new RuntimeException("TF initialization failed");
108     }
109     // The shape of the output is [N, NUM_CLASSES],
110     // where N is the batch size.
111     int numClasses =
112         (int) c.inferenceInterface.graph().operation(
113             outputName).output(0).shape().size(1);
114     Log.i(TAG, "Read " + c.labels.size() +
115             " labels, output layer size is " + numClasses);
116
117     // Ideally, inputSize could have been retrieved
118     // from the shape of the input operation. Alas,
119     // the placeholder node for input in the graphdef
120     // typically used does not specify a shape, so it
121     // must be passed in as a parameter.
122     c.inputSize = inputSize;
123     c.imageMean = imageMean;
124     c.imageStd = imageStd;
125
126     // Pre-allocate buffers.

```

```

127     c.outputNames = new String []{outputName};
128     c.intValues = new int [inputSize * inputSize];
129     c.floatValues = new float [inputSize * inputSize * 3];
130     c.outputs = new float [numClasses];
131
132     return c;
133 }
134
135 @Override
136 public List<Recognition> recognizeImage(final Bitmap bitmap) {
137     // Log this method so that it can be analyzed with systrace.
138     Trace.beginSection("recognizeImage");
139
140     Trace.beginSection("preprocessBitmap");
141     // Preprocess the image data from 0-255 int
142     // to normalized float based
143     // on the provided parameters.
144     bitmap.getPixels(intValues, 0, bitmap.getWidth(), bitmap.getHeight());
145     for (int i = 0; i < intValues.length; ++i) {
146         final int val = intValues[i];
147         floatValues[i * 3 + 0] =
148             (((val >> 16) & 0xFF) - imageMean) / imageStd;
149         floatValues[i * 3 + 1] =
150             (((val >> 8) & 0xFF) - imageMean) / imageStd;
151         floatValues[i * 3 + 2] =
152             ((val & 0xFF) - imageMean) / imageStd;
153     }
154     Trace.endSection();
155
156     // Copy the input data into TensorFlow.
157     Trace.beginSection("fillNodeFloat");
158     inferenceInterface.fillNodeFloat(
159         inputName, new int []
160             {1, inputSize, inputSize, 3}, floatValues);
161     Trace.endSection();
162
163     // Run the inference call.
164     Trace.beginSection("runInference");
165     inferenceInterface.runInference(outputNames);
166     Trace.endSection();
167
168     // Copy the output Tensor back into the output array.
169     Trace.beginSection("readNodeFloat");
170     inferenceInterface.readNodeFloat(outputName, outputs);
171     Trace.endSection();
172
173     // Find the best classifications.
174     PriorityQueue<Recognition> pq =
175         new PriorityQueue<Recognition>(

```

```

177     3,
178     new Comparator<Recognition>() {
179       @Override
180       public int compare(Recognition lhs, Recognition rhs) {
181         // Intentionally reversed to put high confidence at the head of the queue.
182         return Float.compare(rhs.getConfidence(), lhs.getConfidence());
183       }
184     });
185     for (int i = 0; i < outputs.length; ++i) {
186       if (outputs[i] > THRESHOLD) {
187         pq.add(
188           new Recognition(
189             "" + i, labels.size() > i ? labels.get(i) :
190             "unknown", outputs[i], null));
191       }
192     }
193     final ArrayList<Recognition> recognitions = new ArrayList<Recognition>();
194     int recognitionsSize = Math.min(pq.size(), MAX_RESULTS);
195     for (int i = 0; i < recognitionsSize; ++i) {
196       recognitions.add(pq.poll());
197     }
198     Trace.endSection(); // "recognizeImage"
199     return recognitions;
200   }
201
202   @Override
203   public void enableStatLogging(boolean debug) {
204     inferenceInterface.enableStatLogging(debug);
205   }
206
207   @Override
208   public String getStatString() {
209     return inferenceInterface.getStatString();
210   }
211
212   @Override
213   public void close() {
214     inferenceInterface.close();
215   }
216 }
```

9.3.4 InfoChinese.java

```

1 package com.deepcamera.williamyi.deepcamera;
2
3 import android.content.Intent;
4 import android.os.Bundle;
5 import android.support.v7.app.AppCompatActivity;
6 import android.view.View;
```

```
7 import android.widget.Button;
8
9 /**
10 * Created by williamyi on 2017/6/16.
11 * Email: williamyi96@gmail.com
12 */
13
14
15 public class InfoChinese extends AppCompatActivity {
16
17     private Button backCamera;
18     private Button toEnglish;
19     private Button mgithub;
20     private Button mproject;
21     private Button mweibo;
22
23     @Override
24     protected void onCreate(Bundle savedInstanceState) {
25         super.onCreate(savedInstanceState);
26         setContentView(R.layout.info_cn);
27
28         backCamera = (Button) findViewById(R.id.back_camera);
29         toEnglish = (Button) findViewById(R.id.change_to_English);
30         mgithub = (Button) findViewById(R.id.github_location);
31         mproject = (Button) findViewById(R.id.project_location);
32         mweibo = (Button) findViewById(R.id.weibo_location);
33
34         backCamera.setOnClickListener(new View.OnClickListener() {
35             @Override
36             public void onClick(View v) {
37                 Intent intent1 = new Intent();
38                 intent1.setClass(InfoChinese.this, MainActivity.class);
39                 startActivity(intent1);
40             }
41         });
42
43         toEnglish.setOnClickListener(new View.OnClickListener() {
44             @Override
45             public void onClick(View v) {
46                 Intent intent2 = new Intent();
47                 intent2.setClass(InfoChinese.this, InfoEnglish.class);
48                 startActivity(intent2);
49             }
50         });
51
52         mgithub.setOnClickListener(new View.OnClickListener() {
53             @Override
54             public void onClick(View v) {
55                 Intent intent1 = new Intent();
56                 intent1.setClass(InfoChinese.this, ToWebView.class);
```

```

57         intent1.putExtra("mwebsite", "https://github.com/WilliamYi96");
58         startActivityForResult(intent1);
59     }
60 });
61
62 mproject.setOnClickListener(new View.OnClickListener() {
63     @Override
64     public void onClick(View v) {
65         Intent intent2 = new Intent();
66         intent2.setClass(InfoChinese.this, ToWebView.class);
67         intent2.putExtra("mwebsite",
68             "https://github.com/WilliamYi96/DeepCamera");
69         startActivityForResult(intent2);
70     }
71 });
72
73 mweibo.setOnClickListener(new View.OnClickListener() {
74     @Override
75     public void onClick(View v) {
76         Intent intent3 = new Intent();
77         intent3.setClass(InfoChinese.this, ToWebView.class);
78         intent3.putExtra("mwebsite",
79             "http://www.weibo.com/u/5794337545?
80                 from=page_100505_profile&wvr=6&mod=like&is_all=1");
81         startActivityForResult(intent3);
82     }
83 });
84 });
85 }
```

9.3.5 InfoEnglish.java

```

1 package com.deepcamera.williamyi.deepcamera;
2
3 import android.content.Intent;
4 import android.os.Bundle;
5 import android.support.v7.app.AppCompatActivity;
6 import android.view.View;
7 import android.widget.Button;
8
9 /**
10  * Created by williamyi on 2017/6/18.
11  * Email: williamyi96@gmail.com
12  */
13
14
15 public class InfoEnglish extends AppCompatActivity {
16
17     private Button backCamera_en;
```

```
18  private Button toChinese_en;
19
20  private Button mgithub;
21  private Button mproject;
22  private Button mweibo;
23
24  @Override
25  protected void onCreate(Bundle savedInstanceState) {
26      super.onCreate(savedInstanceState);
27      setContentView(R.layout.info_en);
28
29      backCamera_en = (Button) findViewById(R.id.back_camera);
30      toChinese_en = (Button) findViewById(R.id.change_to_Chinese);
31
32      mgithub = (Button) findViewById(R.id.github_location);
33      mproject = (Button) findViewById(R.id.project_location);
34      mweibo = (Button) findViewById(R.id.weibo_location);
35
36      backCamera_en.setOnClickListener(new View.OnClickListener() {
37          @Override
38          public void onClick(View v) {
39              Intent intent1 = new Intent();
40              intent1.setClass(InfoEnglish.this, MainActivity.class);
41              startActivity(intent1);
42          }
43      });
44
45      toChinese_en.setOnClickListener(new View.OnClickListener() {
46          @Override
47          public void onClick(View v) {
48              Intent intent2 = new Intent();
49              intent2.setClass(InfoEnglish.this, InfoChinese.class);
50              startActivity(intent2);
51          }
52      });
53
54      mgithub.setOnClickListener(new View.OnClickListener() {
55          @Override
56          public void onClick(View v) {
57              Intent intent1 = new Intent();
58              intent1.setClass(InfoEnglish.this, ToWebView.class);
59              intent1.putExtra("mwebsite", "https://github.com/WilliamYi96");
60              startActivity(intent1);
61          }
62      });
63
64      mproject.setOnClickListener(new View.OnClickListener() {
65          @Override
66          public void onClick(View v) {
67              Intent intent2 = new Intent();
```

```

68         intent2.setClass(InfoEnglish.this, ToWebView.class);
69         intent2.putExtra("mwebsite", "https://github.com/WilliamYi96/DeepCamera");
70         startActivity(intent2);
71     }
72 });
73
74 mweibo.setOnClickListener(new View.OnClickListener() {
75     @Override
76     public void onClick(View v) {
77         Intent intent3 = new Intent();
78         intent3.setClass(InfoEnglish.this, ToWebView.class);
79         intent3.putExtra("mwebsite",
80                         "http://www.weibo.com/u/5794337545?
81                         from=page_100505_profile&wvr=6&mod=like&is_all=1");
82         startActivity(intent3);
83     }
84 });
85 }
86 }
```

9.3.6 ToWebView.java

```

1 package com.deepcamera.williamyi.deepcamera;
2
3 import android.content.Intent;
4 import android.icu.text.IDNA;
5 import android.os.Bundle;
6 import android.support.v7.app.AppCompatActivity;
7 import android.view.View;
8 import android.webkit.WebView;
9 import android.webkit.WebViewClient;
10 import android.widget.Button;
11
12 /**
13 * Created by williamyi on 2017/6/18.
14 * Email: williamyi96@gmail.com
15 */
16
17 public class ToWebView extends AppCompatActivity {
18
19     @Override
20     protected void onCreate(Bundle savedInstanceState) {
21         super.onCreate(savedInstanceState);
22         setContentView(R.layout.web_view);
23
24         Intent intent = this.getIntent();
25         Bundle bundle = intent.getExtras();
26         String value = bundle.getString("mwebsite");
27 }
```

```
28     WebView webView = (WebView) findViewById(R.id.view_web);
29     webView.getSettings().setJavaScriptEnabled(true);
30     webView.setWebViewClient(new WebViewClient());
31     webView.loadUrl(value);
32 }
33 }
```

9.3.7 Classifier.java

```
1 package com.deepcamera.williamyi.deepcamera;
2
3 /**
4 * Created by williamyi on 2017/6/18.
5 * Email: williamyi96@gmail.com
6 */
7
8
9 import android.graphics.Bitmap;
10 import android.graphics.RectF;
11
12 import java.util.List;
13
14 /**
15 * Generic interface for interacting with different recognition engines.
16 */
17 public interface Classifier {
18 /**
19 * An immutable result returned
20 * by a Classifier describing what was recognized.
21 */
22 public class Recognition {
23 /**
24 * A unique identifier for what has been recognized.
25 * Specific to the class, not the instance of
26 * the object.
27 */
28     private final String id;
29
30 /**
31 * Display name for the recognition.
32 */
33     private final String title;
34
35 /**
36 * A sortable score for how good the recognition is
37 * relative to others. Higher should be better.
38 */
39     private final Float confidence;
```

```
41  /**
42   * Optional location within the source image
43   * for the location of the recognized object.
44   */
45  private RectF location;
46
47  public Recognition(
48      final String id, final String title,
49      final Float confidence, final RectF location) {
50      this.id = id;
51      this.title = title;
52      this.confidence = confidence;
53      this.location = location;
54  }
55
56  public String getId() {
57      return id;
58  }
59
60  public String getTitle() {
61      return title;
62  }
63
64  public Float getConfidence() {
65      return confidence;
66  }
67
68  public RectF getLocation() {
69      return new RectF(location);
70  }
71
72  public void setLocation(RectF location) {
73      this.location = location;
74  }
75
76  @Override
77  public String toString() {
78      String resultString = "";
79      if (id != null) {
80          resultString += "[" + id + "]";
81      }
82      if (title != null) {
83          resultString += title + " ";
84      }
85      if (confidence != null) {
86          resultString += String.format("(%.1f%%)", confidence * 100.0f);
87      }
88  }
```

```

91         if (location != null) {
92             resultString += location + " ";
93         }
94     }
95     return resultString.trim();
96 }
97 }
98 }
99
100    List<Recognition> recognizeImage(Bitmap bitmap);
101
102    void enableStatLogging(final boolean debug);
103
104    String getStatString();
105
106    void close();
107 }
```

9.4 layout

9.4.1 activity-main.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <android.support.v4.widget.DrawerLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     xmlns:tools="http://schemas.android.com/tools"
5     xmlns:app="http://schemas.android.com/apk/res-auto"
6     android:id="@+id/drawer_layout"
7     android:background="@drawable/bg1"
8     android:layout_width="match_parent"
9     android:layout_height="match_parent">
10
11    <FrameLayout
12        android:layout_width="match_parent"
13        android:layout_height="match_parent">
14
15        <android.support.v7.widget.Toolbar
16            android:id="@+id/toolbar"
17            android:layout_width="match_parent"
18            android:layout_height="?attr/actionBarSize"
19            android:background="?attr/colorPrimary"
20            android:layout_marginBottom="8sp"
21            android:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar">
22        </android.support.v7.widget.Toolbar>
23
24        <com.flurgle.camerakit.CameraView
25            android:id="@+id/cameraView"
26            android:layout_width="300dp"
```

```
27     android:layout_height="300dp"
28     android:layout_marginBottom="32sp"
29     android:layout_marginTop="?attr/actionBarSize"
30     android:layout_gravity="center|top" />
31 <LinearLayout
32     android:layout_marginLeft="8sp"
33     android:layout_width="match_parent"
34     android:layout_height="80dp"
35     android:layout_gravity="center|top"
36     android:layout_marginTop="380dp"
37     android:gravity="center"
38     android:orientation="horizontal">
39     <ImageView
40         android:id="@+id/imageViewResult"
41         android:layout_width="75dp"
42         android:layout_height="75dp"
43         android:padding="2dp" />
44     <TextView
45         android:id="@+id/textViewResult"
46         android:layout_width="match_parent"
47         android:layout_height="80dp"
48         android:fadeScrollbars="false"
49         android:maxLines="15"
50         android:scrollbars="vertical"
51         android:gravity="center"
52         android:textColor="@android:color/black" />
53 </LinearLayout>
54 <Button
55     android:id="@+id/btnToggleCamera"
56     android:layout_width="match_parent"
57     android:layout_height="48dp"
58     android:background="@drawable/bg4"
59     android:layout_gravity="bottom|center"
60     android:layout_marginBottom="50dp"
61     android:text="@string/toggle_camera"
62     android:textAllCaps="false"
63     android:textColor="@android:color/black" />
64 <Button
65     android:id="@+id/btnDetectObject"
66     android:layout_width="match_parent"
67     android:layout_height="48dp"
68     android:background="@drawable/bg4"
69     android:layout_gravity="bottom|center"
70     android:text="@string/detect_object"
71     android:textAllCaps="false"
72     android:textColor="@android:color/black"
73     android:visibility="gone" />
74 </FrameLayout>
75
76 <android.support.design.widget.NavigationView
```

```

77     android:id="@+id/nav_view"
78     android:layout_width="match_parent"
79     android:layout_height="match_parent"
80     android:layout_gravity="start"
81     app:menu="@menu/nav_menu"
82     app:headerLayout="@layout/nav_header">
83 </android.support.design.widget.NavigationView>
84 </android.support.v4.widget.DrawerLayout>
```

9.4.2 activity-welcome.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:tools="http://schemas.android.com/tools"
4      android:id="@+id/activity_welcome"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:background="@drawable/welcome"
8      android:orientation="vertical"
9      tools:context="com.deepcamera.williamyi.deepcamera.WelcomeActivity">
10
11  <TextView
12      android:text="DeepCamera"
13      android:layout_width="match_parent"
14      android:layout_height="wrap_content"
15      android:textAlignment="center"
16      android:textColor="@android:color/holo_orange_light"
17      android:textSize="45dp"
18      android:layout_marginTop="100dp"
19      android:layout_alignParentTop="true"
20      android:layout_alignParentLeft="true"
21      android:layout_alignParentStart="true"
22      android:id="@+id/textView1" />
23  <TextView
24      android:text="Android with Tensorflow"
25      android:layout_width="match_parent"
26      android:layout_height="wrap_content"
27      android:textAlignment="center"
28      android:textColor="@android:color/holo_orange_light"
29      android:textSize="35dp"
30      android:layout_marginTop="10dp"
31      android:layout_alignParentTop="true"
32      android:layout_alignParentLeft="true"
33      android:layout_alignParentStart="true"
34      android:id="@+id/textView2" />
35
36  <TextView
37      android:id="@+id/self_info"
38      android:layout_width="match_parent"
```

```
39     android:layout_height="wrap_content"
40     android:text="Created by Kai Yi"
41     android:textAlignment="center"
42     android:textSize="24sp"
43     android:layout_marginTop="30sp"/>/
44
45 <TextView
46     android:layout_width="match_parent"
47     android:layout_height="wrap_content"
48     android:text="Email: williamyi96@gmail.com"
49     android:layout_marginTop="18sp"
50     android:textSize="20sp"
51     android:textAlignment="center"/>/
52 </LinearLayout>
```

9.4.3 info-cn.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:orientation="vertical" android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:id="@+id/info_cn"
7     android:background="@drawable/bg4">
8
9     <LinearLayout
10        android:layout_marginTop="16sp"
11        android:orientation="horizontal"
12        android:layout_width="match_parent"
13        android:layout_height="wrap_content">
14
15         <Button
16             android:id="@+id/back_camera"
17             android:layout_marginLeft="16sp"
18             android:background="@drawable/back2"
19             android:layout_width="42sp"
20             android:layout_height="42sp" />
21         <TextView
22             android:layout_marginBottom="16sp"
23             android:layout_marginTop="8sp"
24             android:layout_marginLeft="40sp"
25             android:layout_width="wrap_content"
26             android:layout_height="wrap_content"
27             android:text="个人简历"
28             android:textAlignment="center"
29             android:textSize="36sp"
30             android:textColor="@android:color/holo_orange_light" />
31         <Button
32             android:id="@+id/change_to_English"
```

```
33         android:layout_marginRight="16sp"
34         android:layout_width="wrap_content"
35         android:layout_height="32sp"
36         android:background="#bfe"
37         android:layout_marginLeft="25sp"
38         android:text="English"/>
39     </LinearLayout>
40
41     <TextView
42         android:layout_marginTop="24sp"
43         android:layout_marginLeft="12sp"
44         android:layout_marginRight="12sp"
45         android:layout_width="wrap_content"
46         android:layout_height="wrap_content"
47         android:text="@string/info_Chinese"
48         android:layout_marginBottom="30sp"
49         android:textSize="18sp"/>
50
51     <TextView
52         android:layout_marginLeft="12sp"
53         android:layout_marginRight="12sp"
54         android:layout_width="wrap_content"
55         android:layout_height="wrap_content"
56         android:textSize="20sp"
57         android:text="@string/slogan_Chinese1"/>
58
59     <TextView
60         android:layout_marginTop="12sp"
61         android:layout_marginLeft="12sp"
62         android:layout_marginRight="12sp"
63         android:layout_width="wrap_content"
64         android:layout_height="wrap_content"
65         android:textSize="20sp"
66         android:text="@string/slogan_Chinese2"/>
67
68     <LinearLayout
69         android:layout_marginLeft="16sp"
70         android:layout_marginRight="16sp"
71         android:layout_marginTop="40sp"
72         android:background="@android:color/holo_blue_dark"
73         android:orientation="horizontal"
74         android:layout_marginBottom="10sp"
75         android:layout_width="match_parent"
76         android:layout_height="wrap_content">
77         <ImageView
78             android:layout_width="48sp"
79             android:layout_height="48sp"
80             android:src="@drawable/github" />
81         <Button
82             android:id="@+id/github_location"
83             android:layout_width="match_parent"
84             android:layout_height="wrap_content"
85             android:background="@android:color/holo_green_light" />
```

```
83         android:text="Github"/>"/>
84     </LinearLayout>
85
86     <LinearLayout
87         android:layout_marginLeft="16sp"
88         android:layout_marginRight="16sp"
89         android:background="@android:color/holo_blue_dark"
90         android:orientation="horizontal"
91         android:layout_marginBottom="10sp"
92         android:layout_width="match_parent"
93         android:layout_height="wrap_content">
94         <ImageView
95             android:layout_width="48sp"
96             android:layout_height="48sp"
97             android:src="@drawable/mproject" />
98
99         <Button
100             android:id="@+id/project_location"
101             android:layout_width="match_parent"
102             android:layout_height="wrap_content"
103             android:background="@android:color/holo_green_light"
104             android:text="Project" />
105     </LinearLayout>
106
107     <LinearLayout
108         android:layout_marginLeft="16sp"
109         android:layout_marginRight="16sp"
110         android:background="@android:color/holo_blue_dark"
111         android:orientation="horizontal"
112         android:layout_width="match_parent"
113         android:layout_height="wrap_content">
114         <ImageView
115             android:layout_width="48sp"
116             android:layout_height="48sp"
117             android:src="@drawable/mweibo" />
118         <Button
119             android:background="@android:color/holo_green_light"
120             android:id="@+id/weibo.location"
121             android:layout_width="match_parent"
122             android:layout_height="wrap_content"
123             android:text="Weibo"/>
124     </LinearLayout>
125
126 </LinearLayout>
```

9.4.4 info-en.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout
```

```
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:orientation="vertical" android:layout_width="match_parent"
5     android:layout_height="match_parent"
6     android:id="@+id/info_en"
7     android:background="@drawable/bg4">
8
9     <LinearLayout
10        android:layout_marginTop="16sp"
11        android:orientation="horizontal"
12        android:layout_width="match_parent"
13        android:layout_height="wrap_content">
14         <Button
15             android:id="@+id/back_camera"
16             android:layout_marginLeft="16sp"
17             android:background="@drawable/back2"
18             android:layout_width="32sp"
19             android:layout_height="32sp" />
20         <TextView
21             android:layout_marginBottom="16sp"
22             android:layout_marginTop="8sp"
23             android:layout_marginLeft="45sp"
24             android:layout_width="wrap_content"
25             android:layout_height="wrap_content"
26             android:text="Personal Info"
27             android:textAlignment="center"
28             android:textSize="32sp"
29             android:textColor="@android:color/holo_orange_light" />
30         <Button
31             android:id="@+id/change_to_Chinese"
32             android:layout_marginRight="16sp"
33             android:layout_width="wrap_content"
34             android:layout_height="32sp"
35             android:background="#bfe"
36             android:layout_marginLeft="45sp"
37             android:text="中文"/>
38     </LinearLayout>
39
40     <TextView
41         android:layout_marginTop="24sp"
42         android:layout_marginLeft="12sp"
43         android:layout_marginRight="12sp"
44         android:layout_width="wrap_content"
45         android:layout_height="wrap_content"
46         android:text="@string/info_English"
47         android:layout_marginBottom="30sp"
48         android:textSize="16sp"/>
49     <TextView
50         android:layout_marginLeft="32sp"
51         android:layout_marginRight="12sp"
52         android:layout_width="wrap_content"
```

```
53     android:layout_height="wrap_content"
54     android:textSize="20sp"
55     android:text="@string/slogan_en1"/>
56 <TextView
57     android:layout_marginLeft="32sp"
58     android:layout_marginRight="12sp"
59     android:layout_width="wrap_content"
60     android:layout_height="wrap_content"
61     android:textSize="20sp"
62     android:text="@string/slogan_en2"/>
63 <LinearLayout
64     android:layout_marginLeft="16sp"
65     android:layout_marginRight="16sp"
66     android:layout_marginTop="40sp"
67     android:background="@android:color/holo_blue_dark"
68     android:orientation="horizontal"
69     android:layout_marginBottom="10sp"
70     android:layout_width="match_parent"
71     android:layout_height="wrap_content">
72     <ImageView
73         android:layout_width="48sp"
74         android:layout_height="48sp"
75         android:src="@drawable/github" />
76
77     <Button
78         android:id="@+id/github_location"
79         android:layout_width="match_parent"
80         android:layout_height="wrap_content"
81         android:background="@android:color/holo_green_light"
82         android:text="Github" />
83 </LinearLayout>
84
85 <LinearLayout
86     android:layout_marginLeft="16sp"
87     android:layout_marginRight="16sp"
88     android:background="@android:color/holo_blue_dark"
89     android:orientation="horizontal"
90     android:layout_marginBottom="10sp"
91     android:layout_width="match_parent"
92     android:layout_height="wrap_content">
93     <ImageView
94         android:layout_width="48sp"
95         android:layout_height="48sp"
96         android:src="@drawable/mproject" />
97
98     <Button
99         android:id="@+id/project_location"
100         android:layout_width="match_parent"
101         android:layout_height="wrap_content"
102         android:background="@android:color/holo_green_light" />
```

```

103         android:text="Project" />
104     </LinearLayout>
105
106     <LinearLayout
107         android:layout_marginLeft="16sp"
108         android:layout_marginRight="16sp"
109         android:background="@android:color/holo_blue_dark"
110         android:orientation="horizontal"
111         android:layout_width="match_parent"
112         android:layout_height="wrap_content">
113         <ImageView
114             android:layout_width="48sp"
115             android:layout_height="48sp"
116             android:src="@drawable/mweibo" />
117         <Button
118             android:background="@android:color/holo_green_light"
119             android:id="@+id/weibo.location"
120             android:layout_width="match_parent"
121             android:layout_height="wrap_content"
122             android:text="Weibo"/>
123     </LinearLayout>
124
125 </LinearLayout>

```

9.4.5 nav-header.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout
3     xmlns:android="http://schemas.android.com/apk/res/android"
4     android:layout_width="match_parent"
5     android:layout_height="180dp"
6     android:background="?attr/colorPrimary"
7     android:padding="10dp">
8
9     <de.hododenhoft.circleimageview.CircleImageView
10        android:id="@+id/icon_image"
11        android:layout_width="70dp"
12        android:layout_height="70dp"
13        android:layout_centerInParent="true"
14        android:src="@drawable/welcome" />
15
16     <!--<TextView-->
17     <!--android:id="@+id/username"-->
18     <!--android:layout_width="wrap_content"-->
19     <!--android:layout_height="wrap_content"-->
20     <!--android:layout_alignParentBottom="true"-->
21     <!--android:text="williamyi96@gmail.com"-->
22     <!--android:textColor="#FFF"-->
23     <!--android:textSize="14sp" /-->

```

```

24    <!--<TextView-->
25        <!--android:id="@+id/mail"-->
26        <!--android:layout_width="wrap_content"-->
27        <!--android:layout_height="wrap_content"-->
28        <!--android:layout_above="@+id/username"-->
29        <!--android:text="William Yi"-->
30        <!--android:textColor="#FFF"-->
31        <!--android:textSize="14sp" /-->
32
33    </RelativeLayout>
34

```

9.4.6 web-view.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      android:orientation="vertical" android:layout_width="match_parent"
4      android:layout_height="match_parent">
5
6      <WebView
7          android:id="@+id/view_web"
8          android:layout_width="match_parent"
9          android:layout_height="match_parent">
10     </WebView>
11 </LinearLayout>

```

9.5 values

9.5.1 colors.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <resources>
3      <color name="colorPrimary">#3F51B5</color>
4      <color name="colorPrimaryDark">#303F9F</color>
5      <color name="colorAccent">#FF4081</color>
6  </resources>

```

9.5.2 strings.xml

```

1  <resources>
2      <string name="app_name">DeepCamera</string>
3      <string name="toggle_camera">切换相机</string>
4      <string name="detect_object">探测物体</string>
5
6      <string name="info_Chinese">
7          易凯， 西安交通大学2015级软件工程专业本科生。

```

```

8     现于智能网络与网络安全教育部重点实验室 ,
9     西安交通大学软件学院创新实验室进行科研与工程训练。
10    目前作为负责人完成国家级大学生创新创业训练课题。
11    主要研究兴趣为深度学习，机器视觉以及增强学习。 </string>
12    <string name="slogan_Chinese1">
13        为天地立心，为生灵立命； </string>
14    <string name="slogan_Chinese2">
15        为往圣继绝学，为万世开太平！ </string>
16    <string name="info_English">
17        William Yi, who majors in software engineering
18            in Xi'an Jiaotong University .
19            He is interested in deep learning ,
20                computer vision and reinforcement learning .
21            And he is the intent of
22                Key Laboratory of Intelligent Network and Network Security ,
23                    Ministry of Education in Xi'an Jiaotong University .
24                    The member of Innovative laboratory in software institute as well .
25                    Besides , He is conducting a national student project.</string>
26    <string name="slogan_en1">Pursue what you what. </string>
27    <string name="slogan_en2">Do only what you can do.</string>
28
29 </resources>

```

9.5.3 styles.xml

```

1 <resources>
2
3     <!-- Base application theme. -->
4     <style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
5         <!-- Customize your theme here. -->
6         <item name="colorPrimary">@color/colorPrimary</item>
7         <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
8         <item name="colorAccent">@color/colorAccent</item>
9     </style>
10
11 </resources>

```

9.6 menu

9.6.1 nav-menu.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <menu
3     xmlns:android="http://schemas.android.com/apk/res/android">
4     <group android:checkableBehavior="single">
5         <item
6             android:id="@+id/nav_call"

```

```

7         android:icon="@drawable/nav_call"
8         android:title="Call" />
9     <item
10        android:id="@+id/nav_friends"
11        android:icon="@drawable/nav_friends"
12        android:title="Friends" />
13     <item
14        android:id="@+id/nav_location"
15        android:icon="@drawable/nav_location"
16        android:title="Location" />
17     <item
18        android:id="@+id/nav_mail"
19        android:icon="@drawable/nav_mail"
20        android:title="Mail" />
21     <item
22        android:id="@+id/nav_task"
23        android:icon="@drawable/nav_task"
24        android:title="Tasks" />
25   </group>
26 </menu>
```

9.6.2 toolbar.xml

```

1 <?xml version="1.0" encoding="utf-8"?>
2 <menu xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto">
4   <item
5     android:id="@+id/backup"
6     android:icon="@drawable/ic_backup"
7     android:title="Backup"
8     app:showAsAction="always"/>
9   <item
10    android:id="@+id/delete"
11    android:icon="@drawable/ic_delete"
12    android:title="Delete"
13    app:showAsAction="ifRoom"/>
14   <item
15    android:id="@+id/information"
16    android:icon="@drawable/ic_settings"
17    android:title="Settings"
18    app:showAsAction="never"/>
19 </menu>
```