

# 使用TensorFlow搭建智能开发系统 自动生成App UI代码

郝振明 / 普元信息 移动产品部

1. 我们的现状与期望
2. 我们的初级探索及建议
3. 智能开发系统的自建之路
4. 未完待续

# 我们的现状与期望



- 让初级开发人员具备专家80%的能力
- 让AI辅助移动App的研发工作

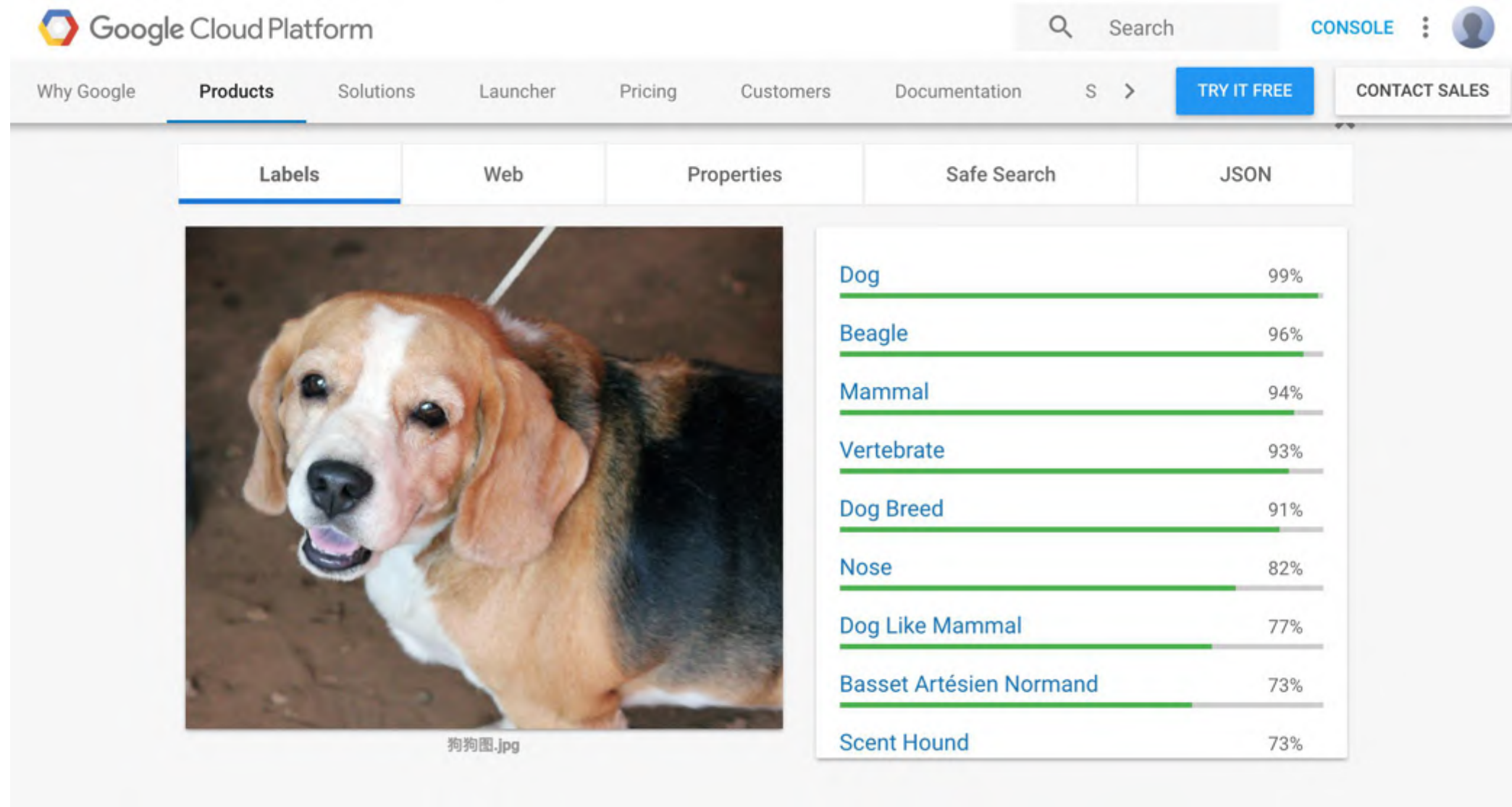
# 我们的现状与期望



切入点：从设计稿（或者App截图）到App前端代码

1. 我们的现状与期望
2. 我们的初级探索及建议
3. 智能开发系统的自建之路
4. 未完待续

# 公有云服务的尝试

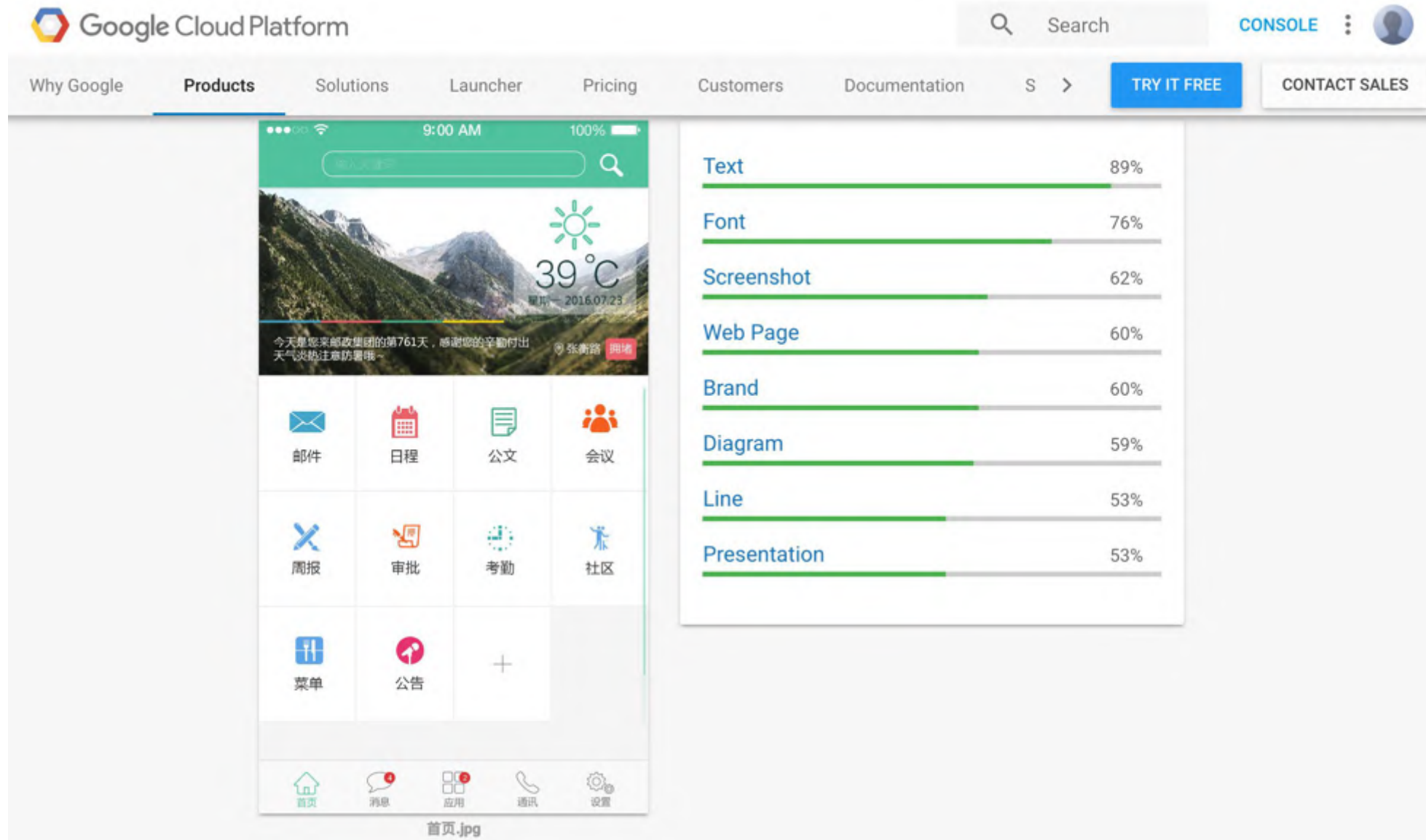


The screenshot shows the Google Cloud Platform interface for image labeling. On the left, there is a photo of a beagle dog. Below the photo is the filename "狗狗图.jpg". On the right, there is a list of labels with their corresponding confidence percentages, each accompanied by a green progress bar. The labels and their percentages are:

Label	Confidence
Dog	99%
Beagle	96%
Mammal	94%
Vertebrate	93%
Dog Breed	91%
Nose	82%
Dog Like Mammal	77%
Basset Artésien Normand	73%
Scent Hound	73%

- 准确识别狗，并且能够识别到品种；（能够识别不同设计就好了）
- 有“鼻子”有眼的，看上去很美，（能够识别设计里的UI控件就够了）

# 公有云服务的尝试



- 实际上，效果如上图。





# 公有云服务的尝试

Try Train

Watson sees...



[JSON](#)

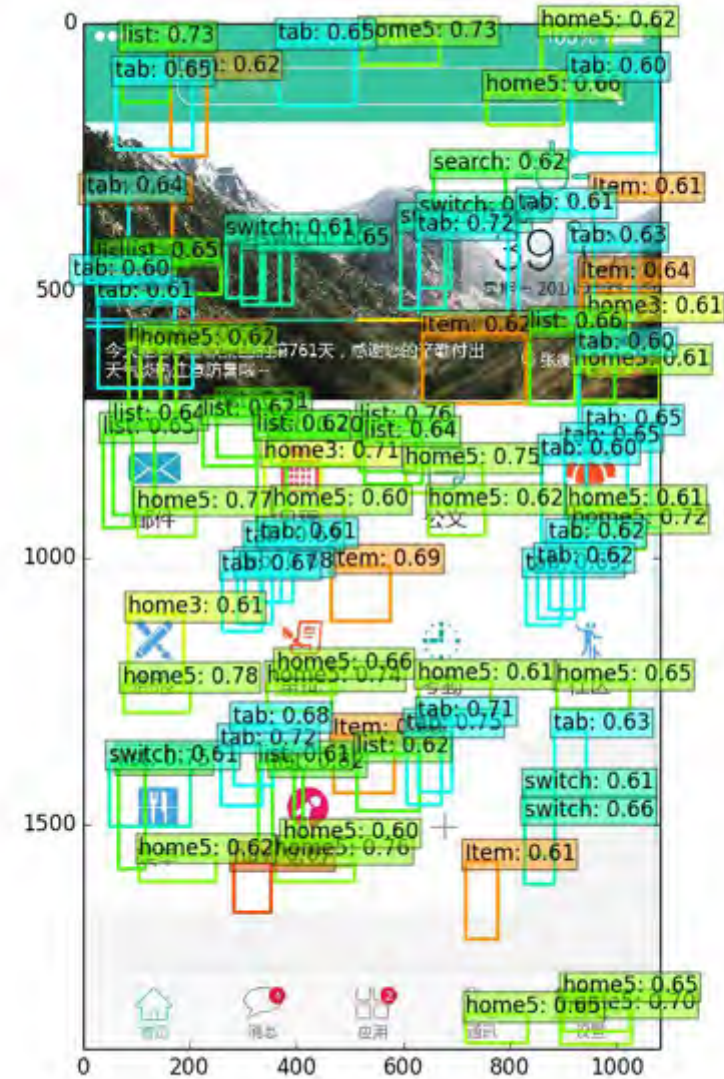
Classes	Score
window	0.97  1
video display	0.97  1
electronic device	0.97  1
device	0.97  1

- 换了另外一家的效果，依旧无法使用。



- 广泛的认知，无法满足垂直的需要
- 我们需要的AI是：宁可不知狗为何物，也必须理解设计懂得App开发
- 我们要做：“生活不能自理的程序员” 😊

# 搭建私有云，使用已有模型



- 普遍的目标检测都是无嵌套结构的认知
- 而App 的UI是一个结构化数据，存在嵌套关系、层级关系
- 上图中的目标检测对于生成代码没有任何帮助

1. 我们的现状与期望
2. 我们的初级探索及建议
3. 智能开发系统的自建之路
4. 未完待续

# 训练的方向，决定了结果

- 专注：训练一个只懂移动UI的程序员
- 学习移动App UI 的设计稿或者截图
- 不学习Web UI 或者GUI的设计稿
- 坚决不学习常规事物 😊

# 调戏一把

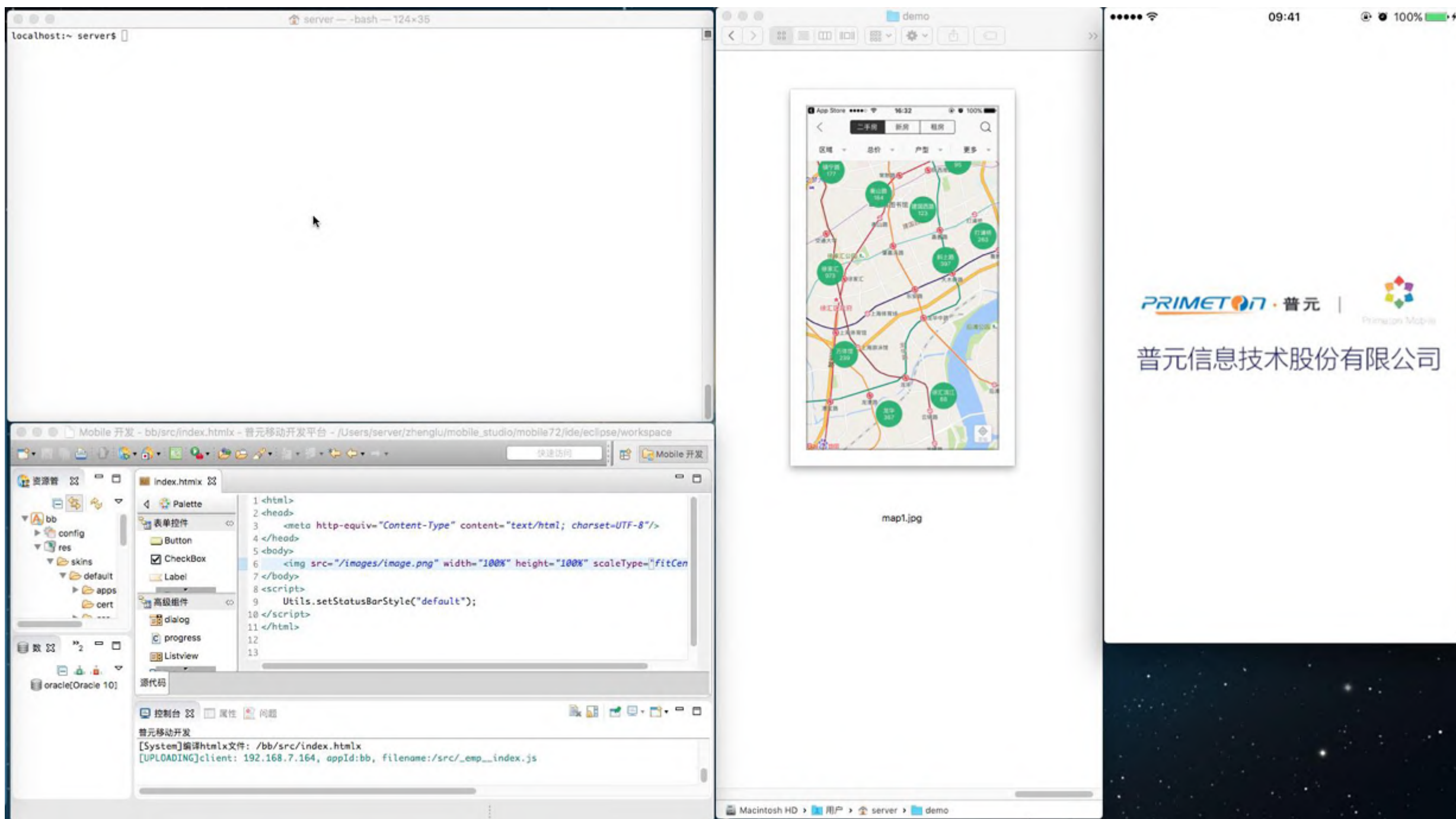


```
portal (score = 0.72156)
gesture (score = 0.10528)
map (score = 0.08253)
mainpage (score = 0.02903)
login (score = 0.02270)
gridview (score = 0.01688)
calendar (score = 0.01638)
list (score = 0.00532)
setting (score = 0.00032)
```

- 同样的这只狗，系统把它当成了欢迎页而不是狗，不错。



# 先看一段视频



# 工程师一般是如何工作？



- 拿到设计，先判断一下是个什么类型界面
- 大概由几部分组成，什么都有哪些控件
- 编写代码
- 微调代码，调试，根据与设计图的差距进行调整

# 采用分类，确定界面类型






# 分类的效果

普元移动平  
移动平台智能开发系统使用  
选择一张APP设计图或是AP

login2.jpg  
(78.44 KB)

匹配结果



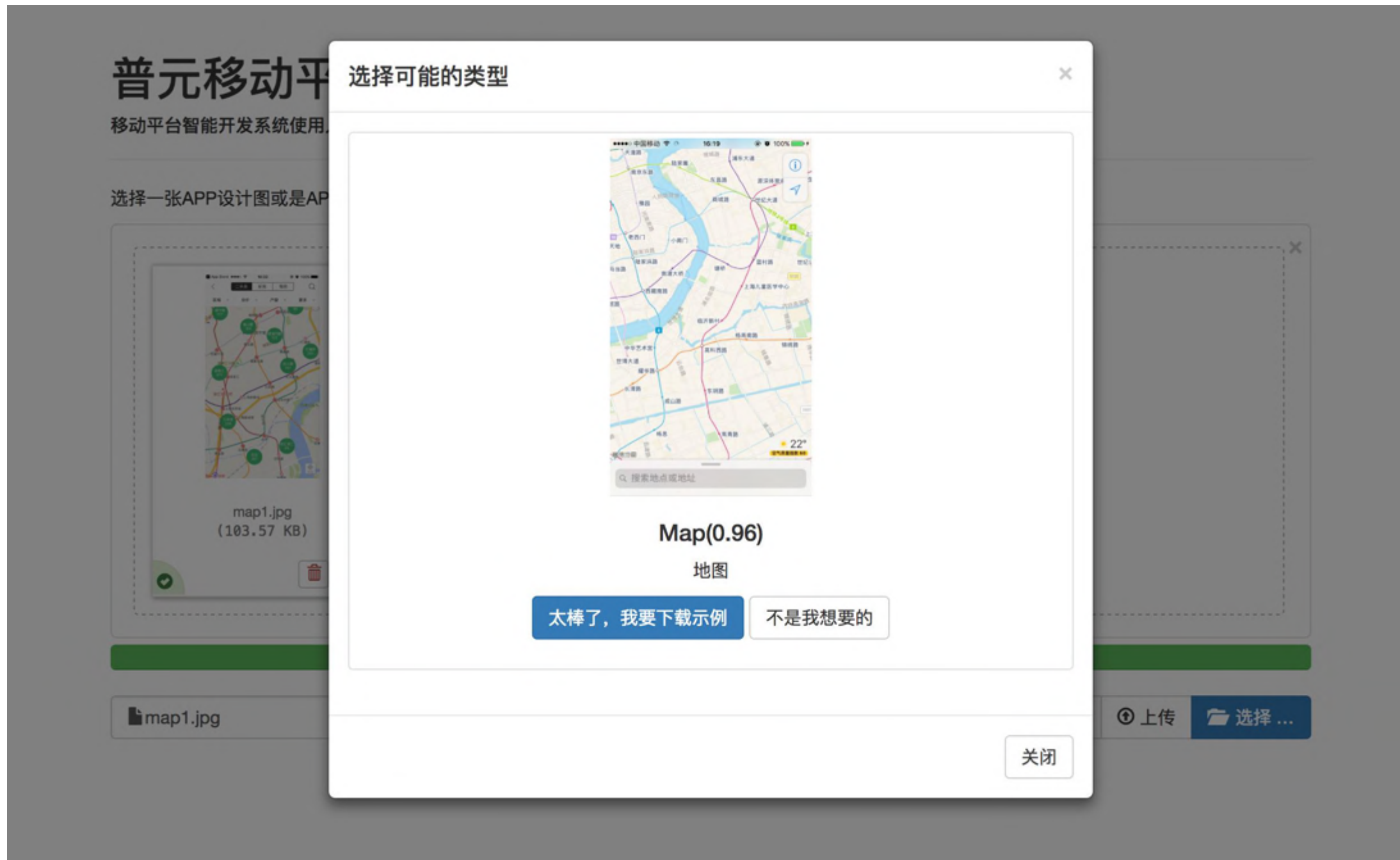
**Login(0.92)**  
登陆页面包括用户名输入框，密码输入框和登录按钮等

太棒了，我要下载示例    不是我想要的

关闭

上传    选择 ...

# 分类的效果

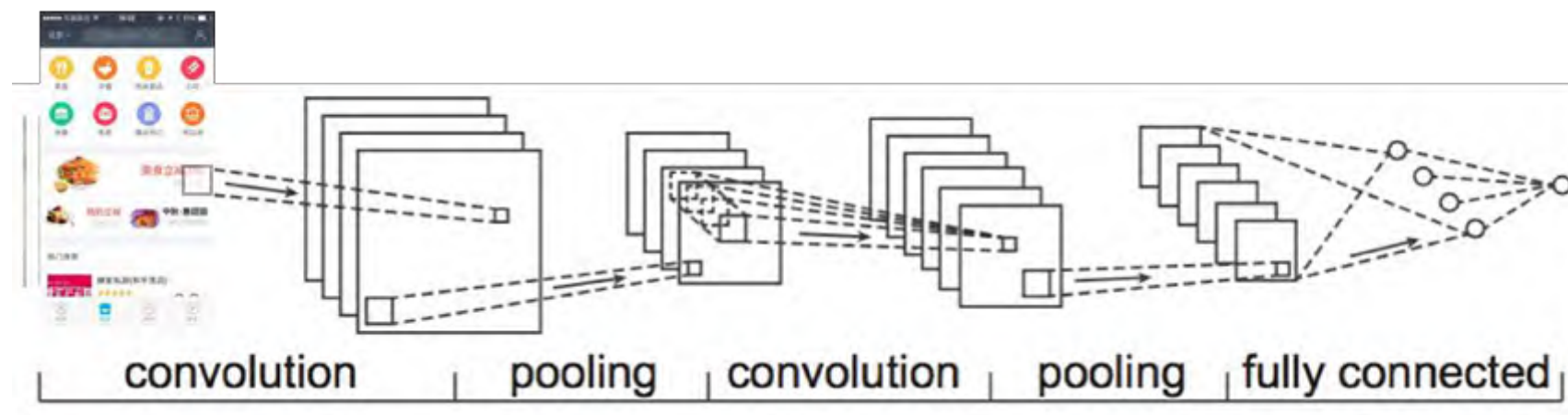


# 在分类上我们的一些工作

- 迁移学习 CNN ( Inception V3 or VGG 16 ) : 模型微调, 应用到我们的数据集上
- 分类器的选择: Softmax更适合, 放弃Logistic
- 数据集: 数据集的积累, 并考虑结构的依赖, 大量引入灰度图片
- 应用: 采用原色及灰度两种, 二者取其高
- 低概率: 引入人工干预, 每个使用者都是监督者
- 闭环: 数据集及分类的积累

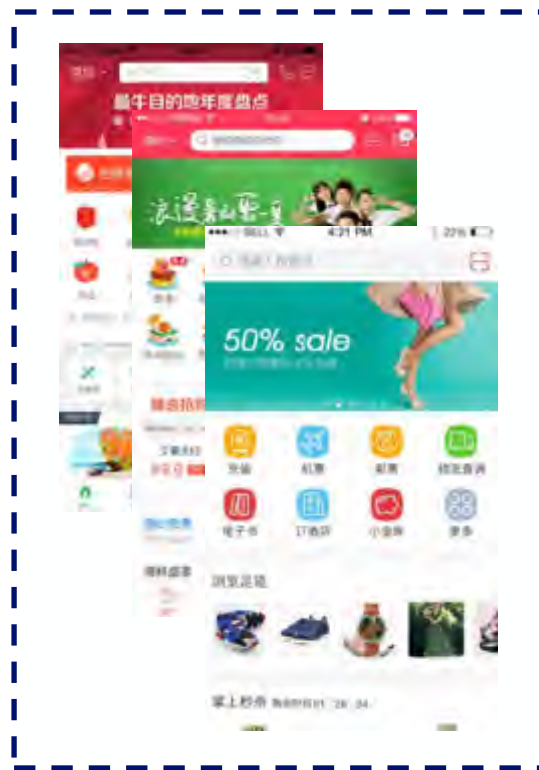
# 迁移学习

- 选用CNN模型，前面各层的参数都不变，而只训练最后一层。
- 将自己的训练集中的每张图像输入到CNN网络中，得到2048个维度的向量特征。
- 利用这些特征来训练softmax的多分类器



# 数据集

- 通过移动平台在各个项目组的实施，收集到各类移动界面图片，我们利用这些图片作为移动的数据集



mainpage



login



map



# 同源灰图训练，提高正确率



```
mainpage (score = 0.74868)
list (score = 0.10108)
map (score = 0.06959)
gridview (score = 0.06808)
setting (score = 0.00396)
calendar (score = 0.00330)
gesture (score = 0.00298)
portal (score = 0.00116)
login (score = 0.00038)
```



```
mainpage (score = 0.91191)
gridview (score = 0.06284)
list (score = 0.01765)
map (score = 0.00205)
calendar (score = 0.00220)
gesture (score = 0.00206)
portal (score = 0.00027)
setting (score = 0.00016)
login (score = 0.00007)
```

# 每一个使用者都成为机器学习的监督者



# 闭环：数据集及分类的积累





# 采用多级目标检测，确定UI元素



# 目标检测的效果



# 在目标检测上我们的一些工作

- 多级Faster-RCNN，避免单一的Faster-RCNN导致目标检测结果叠加
- 阀门（0.8）控制退出，组件优先。
- 加大数据集的积累，避免天窗效果（传统的目标检测没有这个问题）。
- 参数的调整：eg. ANCHOR\_SCALES（无须太小）、BATCH\_SIZE（与数据集数量匹配，不易过大过小）、RPN\_NMS\_THRSH（模型训练越好，越可以提高，但不易过高）等等

# 天窗效果





# 基于DSL语言，生成代码

- Android Java 与 iOS Object-C的代码复杂度
- iOS / Android设计UI趋同，跨平台诉求
- 普元移动平台已经采用DSL层 😊
- 天窗的处理

# 原生语言的代码复杂度



```
主页3_android代码.xml * UNREGISTERED
主页3_android代码.xml *
1 <RelativeLayout xmlns:android="http://schemas.android.
  com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:layout_width="match_parent"
4   android:layout_height="match_parent"
5   tools:context="com.example.testing.MainActivity" >
6
7   <LinearLayout
8     android:layout_width="match_parent"
9     android:layout_height="match_parent"
10    android:orientation="vertical" >
11
12    <RelativeLayout
13      android:layout_width="match_parent"
14      android:layout_height="44dp"
15      android:background="#36C49E" >
16
17      <EditText
18        android:layout_width="250dp"
19        android:layout_height="35dp"
20        android:layout_centerHorizontal="true"
21        android:layout_centerVertical="true"
22        android:ems="10"
23        android:hint="输入关键词" >
24
25        <requestFocus />
26      </EditText>
27
28      <ImageView
29        android:id="@+id/imageView1"
30        android:layout_width="30dp"
31        android:layout_height="30dp"

```



# 天窗的处理：占位

```
5 <body>
6
7 <div width="100%" height="39" vAlign="middle" hAlign="center">
8   <input type="text" height="31" width="320" emptyText="search" backgroundColor="#dddddd" />
9 </div>
10 <div width="100%" height="98" vAlign="middle" hAlign="center">
11   
12 </div>
13 <div width="100%" height="165" vAlign="middle" hAlign="center">
14   <grid width="100%" height="100%" column="5" row="2" itemHeight="82" itemWidth="72"
15     itemImageHeight="27" itemImageWidth="27" textSize="14">
16     <griditem icon="/apps/1.png" label="标签" tag="name" />
17     <griditem icon="/apps/2.png" label="标签" tag="name" />
18     <griditem icon="/apps/3.png" label="标签" tag="name" />
19     <griditem icon="/apps/4.png" label="标签" tag="name" />
20     <griditem icon="/apps/5.png" label="标签" tag="name" />
21     <griditem icon="/apps/6.png" label="标签" tag="name" />
22     <griditem icon="/apps/7.png" label="标签" tag="name" />
23     <griditem icon="/apps/8.png" label="标签" tag="name" />
24     <griditem icon="/apps/9.png" label="标签" tag="name" />
25     <griditem icon="/apps/0.png" label="标签" tag="name" />
26   </grid>
27 </div>
28 <div width="100%" height="100%" vAlign="middle" hAlign="center">
29   <div width="100%" height="100%" hAlign="center" vAlign="middle" layout="HBox" borderWidth="1"
30     borderColor="#dddddd">
31     <input type="label" value="对不起，我暂时不认识" fontSize="20" />
32   </div>
33 </div>
34 <div width="100%" height="41" vAlign="middle" hAlign="center">
35   <div width="100%" height="100%" vAlign="middle" hAlign="center" layout="VBox">
36     <div backgroundColor="#aaaaaa" height="1" width="100%" />
37     <tab width="100%" height="100%" itemHeight="41" enableScroll="false">
38       <tabitem text="菜单" icon="/tab/1.png" textSize="12" iconWidth="24" iconHeight="24">
39         </tabitem>
40       <tabitem text="菜单" icon="/tab/2.png" textSize="12" iconWidth="24" iconHeight="24">
41         </tabitem>
42       <tabitem text="菜单" icon="/tab/0.png" textSize="12" iconWidth="24" iconHeight="24">
43         </tabitem>
44       <tabitem text="菜单" icon="/tab/1.png" textSize="12" iconWidth="24" iconHeight="24">
45         </tabitem>
46       <tabitem text="菜单" icon="/tab/2.png" textSize="12" iconWidth="24" iconHeight="24">
47         </tabitem>
48     </tab>
49   </div>
50 </div>
```

生成的代码



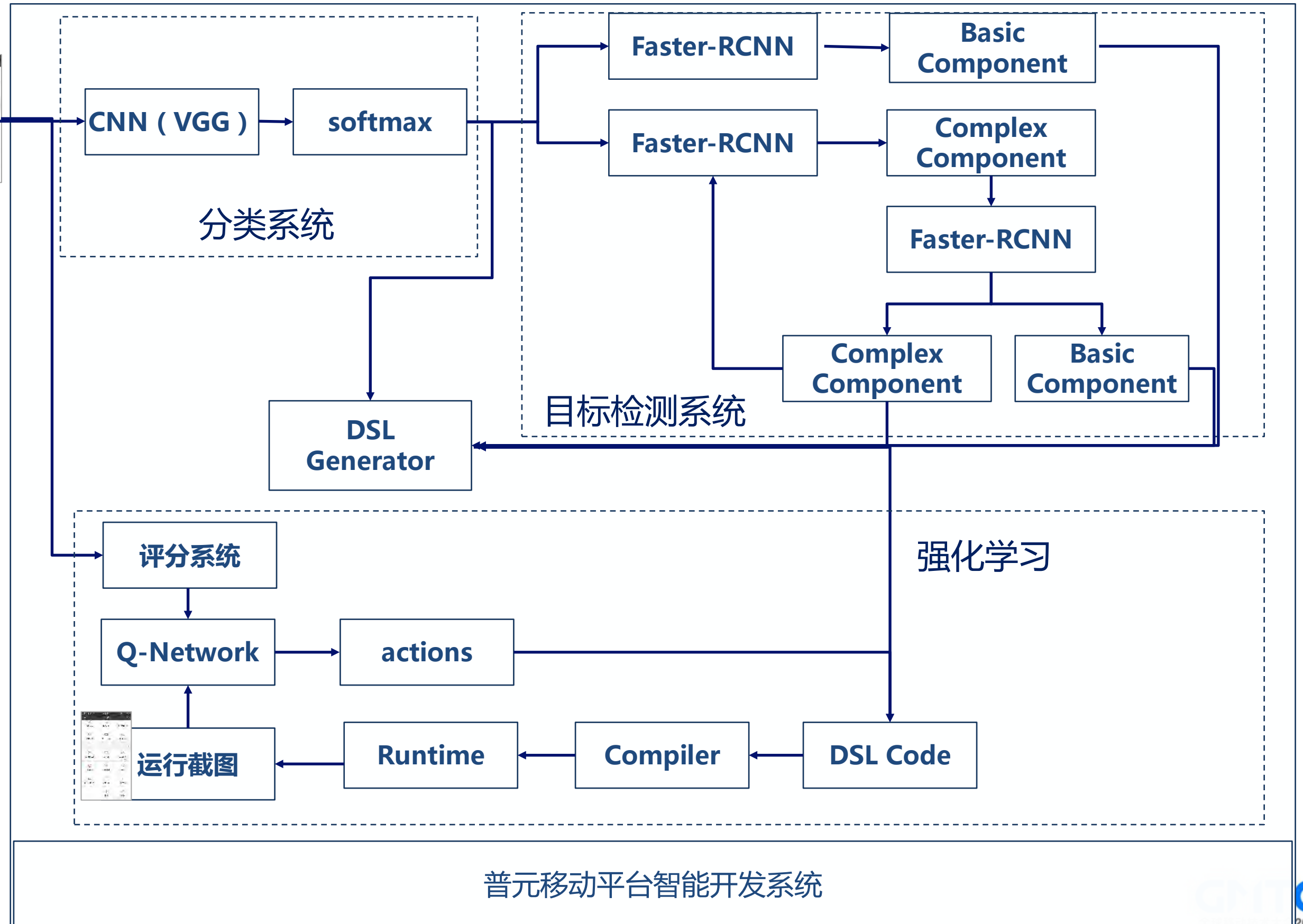
生成代码的运行截图

# 强化学习（RL）对于代码微调

- Actions 只针对component属性调整，不对代码结构调整
- 评分系统依赖原有输入（设计稿或者截图）
- Environment采用生成代码后的运行截屏
- 结合区域性评分（结合Faster-RCNN过程的切图）
- 暂时只针对位置、字体大小等



# 总体结构



1. 我们的现状与期望
2. 我们的初级探索及建议
3. 智能开发系统的自建之路
4. 未完待续

# 关于文字的处理

- 拟对接成熟的OCR或者云服务。

# 关于图片及icon的处理

- 一种是设计师提供设计图，并有小的切图，这个处理比较简单。
- 另外一种是无原图或者需要对接图库的（图库规模不大的情况下），拟采用CNN+Softmax做一次分类，没一张图片一张分类

# AI与移动结合的其他实践：智能化CUI



# AI与移动结合的其他实践：智能推荐及辅助决策



# 欢迎一起探讨



THANKS!

