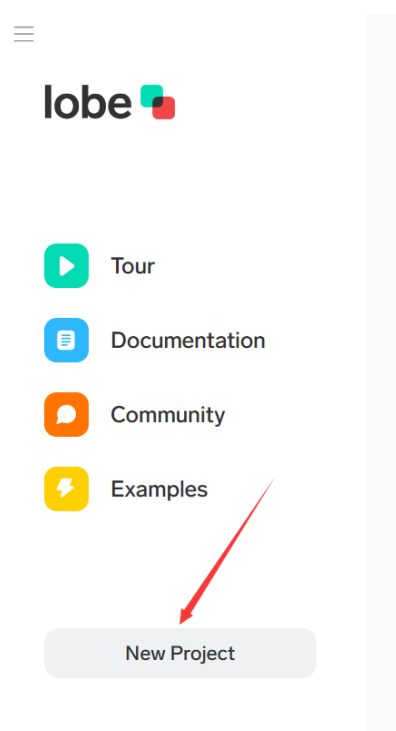
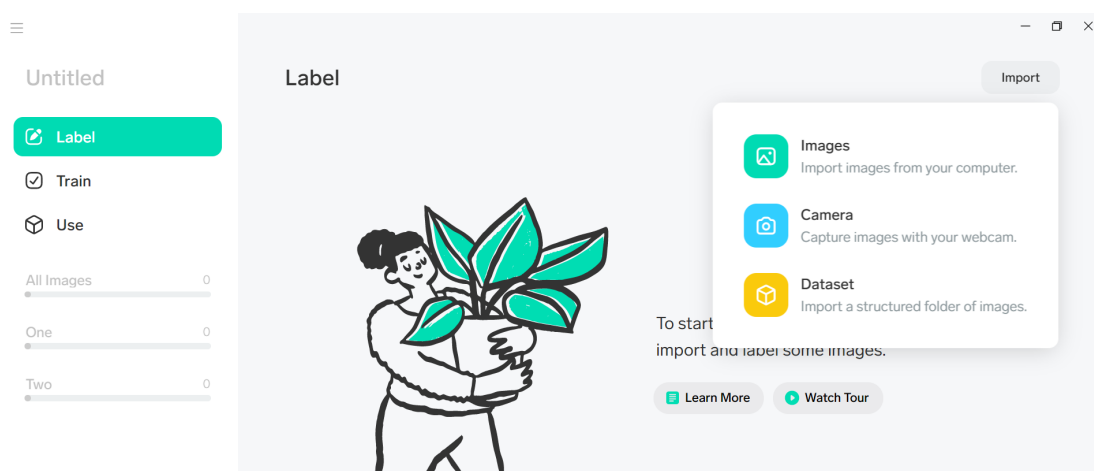


1. Lobe 模型的建立

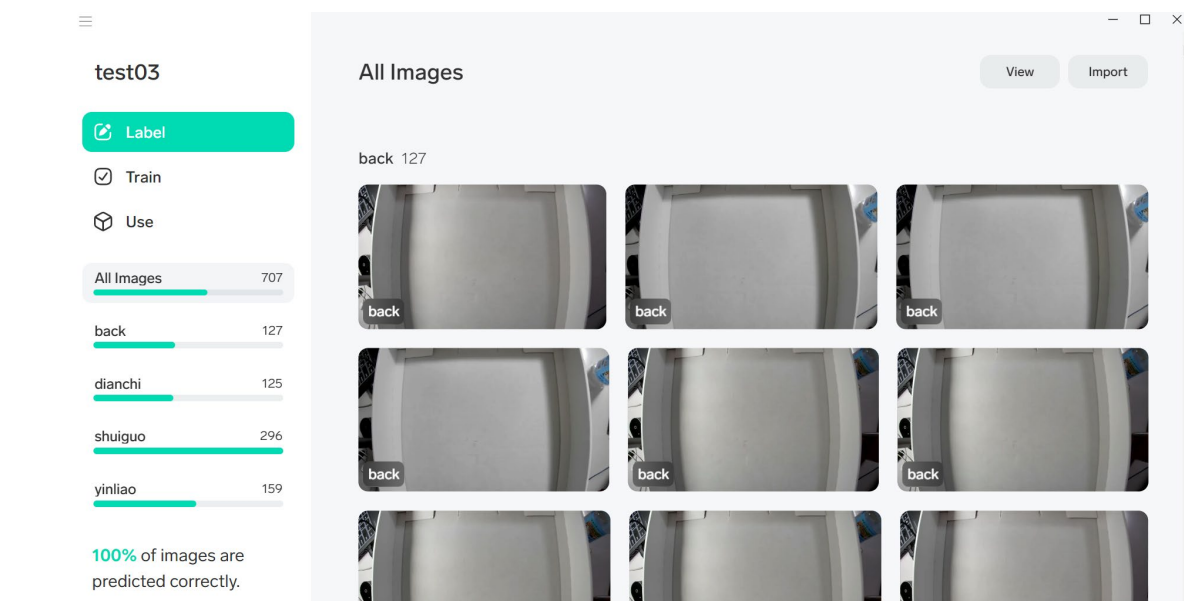


照片可以选择已有文件夹或者直接调用摄像头进行一张一张的拍摄，设立好标签然后就可以开始训练了（如果没反应可以重启软件）。如下图。



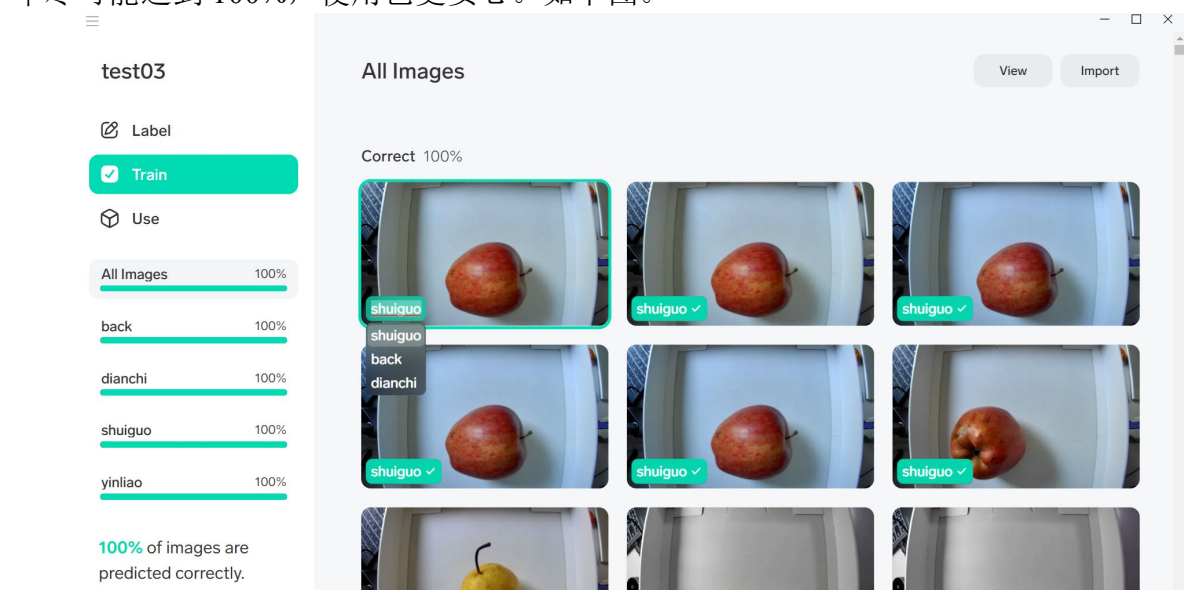
要进行多个物品的识别，这里给出一种方案，即将背景也作为一个标签单独学习，其他物品在背景中再一次进行分类。

训练完成以后可以通过照片或者摄像头进行检验（一旦卡死重启即可），如果检验结果与事实不符还可以在下方改正，待自动训练完成后再尝试。如下图。



这里背景使用了 127 张图片，其中包含了普通光源和其他光源，可以增强模型的抗干扰能力。每种物品大概都在 50-100 张图片（识别效果还不错），不同的位置摆放，不同的姿态，不同的光照都可以多次拍摄，按自己需求拍摄。

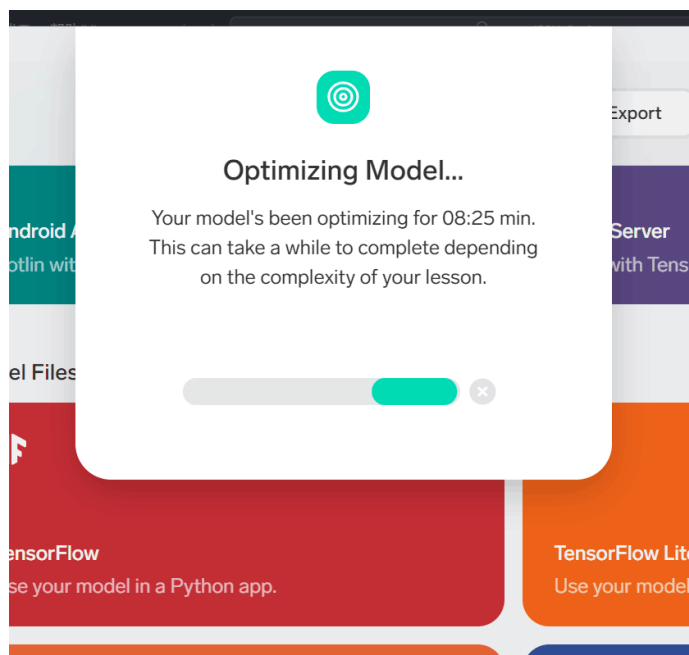
等待训练完成后，可能会出现错误的识别结果而显示红色，可以手动一张一张进行标签的修改，也可以选中一张按 **shift** 进行多选后，修改其中一张照片的标签即可。每次修改都会自动重新对模型进行修正，这里建议多花些时间让识别率尽可能达到 100%，使用也更安心。如下图。



训练结束后，最好到 Use 界面使用图片或者摄像头（推荐）进行检验，如果出现识别错误，也可以在左下角手动纠正，多次检验，可能要重新进行上一步的操作直到识别率达到 100% 即可。达到自己的期望后可以导出模型。

2. Lobe 模型的导出和使用（Python）

导出的形式一般为 TensorFlow 或 TensorFlow Lite（推荐，感觉一般不会出现问題）均可，点击后可以进行模型优化，等待模型导出完成。



使用 Python 语言需要预先导入 lobe 库，复制下列命令在终端中运行：
`pip install lobe -i https://pypi.tuna.tsinghua.edu.cn/simple`

输入下列代码，`ImageModel.load()` 中为导出模型的文件夹的路径，`predict_from_file()` 中为测试图片的路径。运行本代码观察输出即可。

```
from lobe import ImageModel

model = ImageModel.load('/home/duan/Desktop/Untitled TFLite')

result = model.predict_from_file('/home/duan/Desktop/pytest/01.jpg')
a = result.prediction

print(a)
```

3. 使用摄像头进行实际检验

摄像头的调用需要 OpenCV 库，在终端中输入下列命令：

`pip install opencv-python -i https://pypi.tuna.tsinghua.edu.cn/simple`
安装完成后输入下列代码检测自己的摄像头是否可以正常打开。

```

import cv2
cap = cv2.VideoCapture(0)  # 调用摄像头'0'一般是打开电脑自带摄像头,
                           # '1'是打开外部摄像头(只有一个摄像头的情况)

# 显示图像
while True:
    ret, frame = cap.read()  # 读取图像(frame就是读取的视频帧)
    cv2.imshow("frame", frame)

    input_str = cv2.waitKey(20)
    if input_str == ord('q'):  # 鼠标点击视频画面输入字符q 结束程序

        break

cap.release()  # 释放摄像头
cv2.destroyAllWindows()  # 销毁窗口

```

出现报错可自行百度解决。如果正常打开无误可以进行下一步操作：

```

1
2 import cv2
3
4 from lobe import ImageModel
5
6 model = ImageModel.load('E:/python_documents/test03/test03 TFLite')
7 # 该路径为导出的模型文件路径
8 font_style = cv2.FONT_HERSHEY_TRIPLEX # 设置字体样式
9
10
11 capture = cv2.VideoCapture(1)
12
13 def predict():
14     retval, image = capture.read()
15     cv2.imwrite("D:\photo\zhaopian.jpg", image) # 存储图像
16     # 上面的photo文件夹可以自行创建,可以将保存的照片随意取名
17
18     result = model.predict_from_file('D:\photo\zhaopian.jpg')
19     a = result.prediction
20     cv2.putText (image, a, (0, 50), font_style, 1,(0, 255, 255))
21     # 在视频界面显示识别结果
22     cv2.imshow("video", image)
23
24
25
26 while(capture.isOpened()):
27     predict()
28
29     key = cv2.waitKey(1)
30     if key == 32: # 按空格键退出程序
31         break
32
33 capture.release()
34 cv2.destroyAllWindows()
35
36
37

```

上述代码可能与前图稍有不同，注意即可。运行程序，即可看到识别结果，按空格键退出程序。

