Diabetic Retinopathy Detection
糖尿病视网膜病变

2019年12月

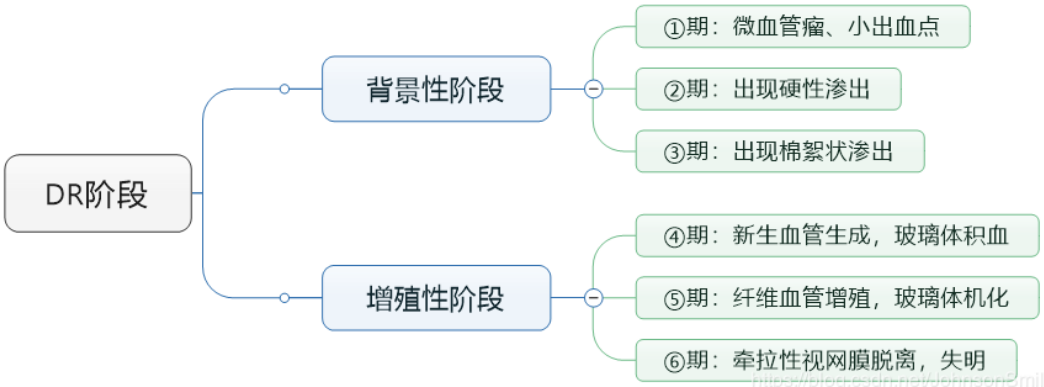
Group 25

上海交通大學
SHANGHAI JIAO TONG UNIVERSITY

目录 Contents

Background

- 糖尿病视网膜病变(DR)是糖尿病最常见的微血管并发症之一。
- 临床表现：视物模糊、视力下降失明等，更严重的致肾功能衰竭及死亡。
- 常见症状：微血管瘤、出血斑点、硬性渗出，除此之外还有棉绒斑、静脉串珠状、视网膜内微血管异常(IRMA)以及黄斑水肿等。
- DR 分级：



Task

- 通过糖尿病视网膜病变的三种主要症状对眼球影像集进行学习，并对待分级的测试图片集进行预测。
- 在此之前需要对眼球影像集进行一些预处理，包括：根据所给的表格对数据集中的图片进行分级，并选取分好类的训练集、验证集和测试集图片；以及对图像预处理的去中心化等操作。

Approach-1

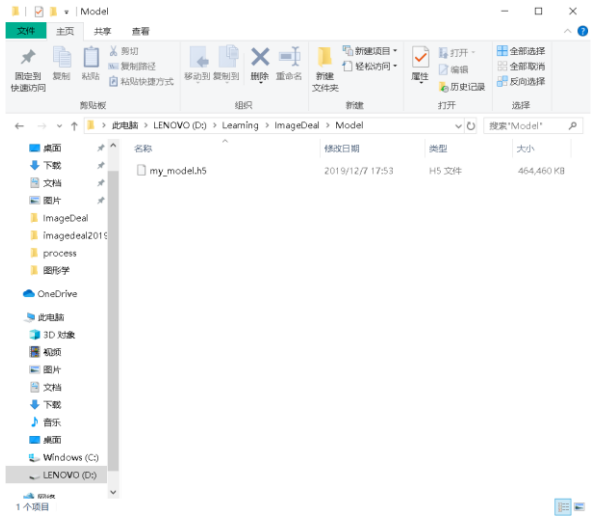
1. **图像预处理：利用Keras内置处理方案**

- 医学图像的特点：灰度上的模糊性（噪声的干扰以及内部组织在成像过程中的相似性）、局部效应、不确定性特点
- 医学图像的特征：颜色、纹理、形状.
- train_datagen = ImageDataGenerator(rescale=1./255,rotation_range=180,horizontal_flip=True,vertical_flip=True,samplewise_center=True,fill_mode='constant')
- 利用旋转、横向翻转、纵向翻转的方式扩增训练集；利用samplewise_center方式使得训练集内图片的形式趋同，便于训练
- 验证集同理。

Approach-1

2. 运用Keras搭建框架学习训练模型

- Keras利用tensorflow作为底层
- 训练模型——train
- 验证模型寻找最优——validation
- 得到HDF5格式的模型




```
if __name__ == '__main__':
    model = Sequential()
    model.add(Conv2D(64, 3, activation='relu', input_shape=(2000,2000,3)))
    model.add(MaxPooling2D(pool_size=(2, 2)))
    model.add(Conv2D(64, 3, activation='relu'))
    model.add(MaxPooling2D(pool_size=(2, 2)))
    model.add(Flatten())
    model.add(Dense(64, activation='relu'))
    model.add(Dropout(0.5))
    model.add(Dense(1, activation='sigmoid'))
```

```
    model.compile(loss='binary_crossentropy',
                  optimizer='rmsprop',
                  metrics=['accuracy'])
    train_datagen = ImageDataGenerator(rescale=1./255)
    test_datagen = ImageDataGenerator(rescale=1./255)
    train_generator = train_datagen.flow_from_directory('train',
                                                        target_size=(2000,2000),
                                                        batch_size=2,
                                                        class_mode='sparse')
```

```
    model.fit_generator(train_generator,
                        steps_per_epoch=100,
                        epochs=50,
                        validation_data=validation_generator,
                        validation_steps=50)
    model.save('D:\Learning\ImageDeal\Model\my_model.h5')
    read_test('D:\Learning\ImageDeal\annotation4\test')  #测试集验证,批量
```

Approach-1

3.* 模型内部用到的提高性能方法

- 验证集：加入验证集使得训练的性能得以保证
- ReLU函数：更好的线性Activation Function，优于传统sigmoid
- RMSProp：能随着训练的进行改变学习率
- Dropout：使得模型更健壮，避免overfitting
- batch_size：用较小的batch收敛模型，占用较小内存，但导致准确率有所下降

Approach-1

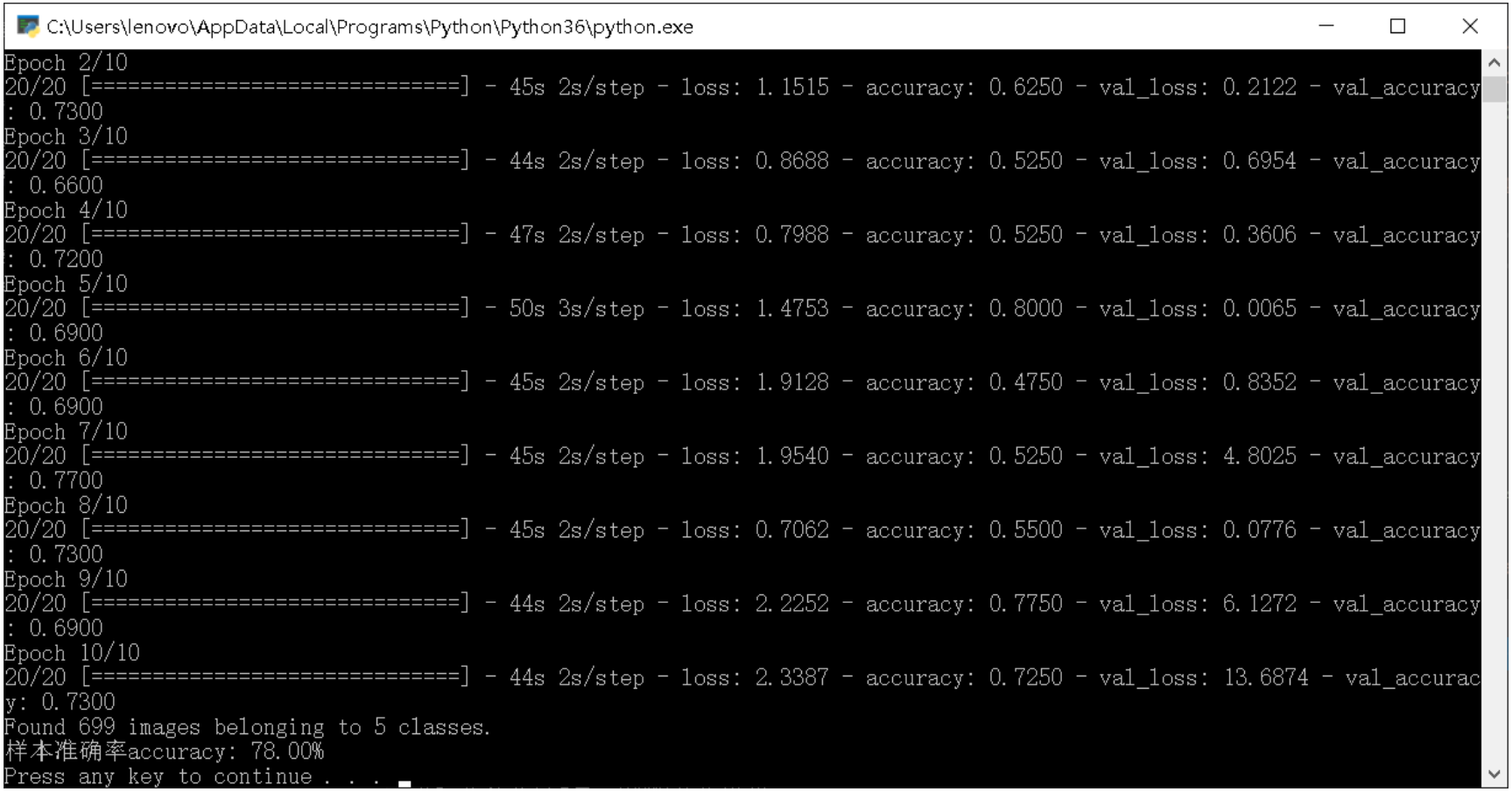
4. **导入训练好的模型对 test 集进行预测**

- 导入之前训练好的最优模型——Load_model
- 模型测试——test

```
test_datagen = ImageDataGenerator(rescale=1. / 255)
test_generator = test_datagen.flow_from_directory('test',
                                                  target_size=(600,400),
                                                  batch_size=2,
                                                  class_mode='sparse')
model = load_model('D:\Learning\ImageDeal\Model\my_model.h5')
score = model.evaluate_generator(test_generator,steps=50)
print("样本准确率%s: %.2f%%" % (model.metrics_names[1], score[1] * 100))
```

Outcome

- 运用 Keras 神经网络对图片集进行训练和验证，以此来获得不同的模型进行比较，最终得到最好的训练模型以及test准确率是78.00%。



Approach-1

5. **可改进及优化的部分**

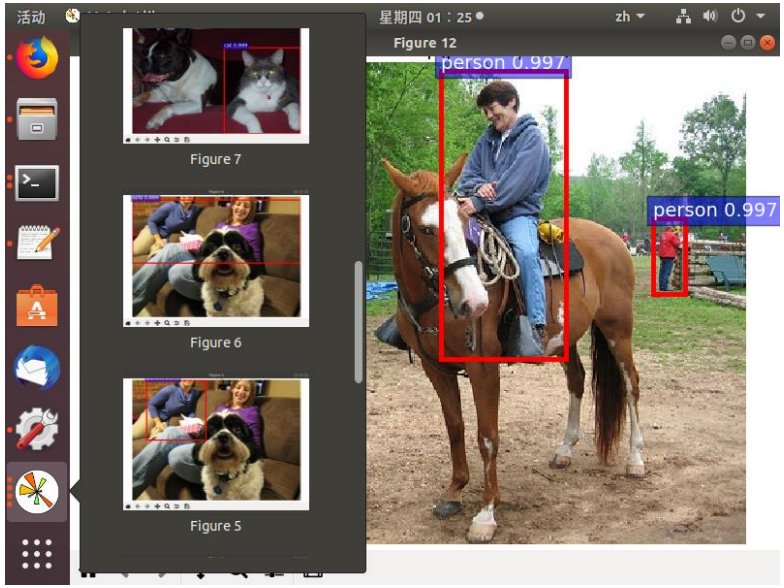
- 尝试使用更好的GPU训练模型
- 由于tensorflow的限制，导入的图像需要有固定的大小和比例，但是提供的视网膜照片有3:2和1:1的图像，导致部分图像被扭曲，影响准确率
- => 可行的改进方式:
- 1. 将不同比例的图片分开训练：由于数据集只有1w多张图片，可能会使得训练效果不佳
- 2.使用别的模型而不仅仅是简单的线性(Sequential)模型
- 3. 利用SPP-net（金字塔池化）使得CNN能输入不同尺寸图像

Approach-2

1. 了解faster-rcnn并运行demo

- Faster-rcnn是利用已标注好的图像进行特征抽取，并在测试图像中分类得到ROI区域的神经网络
- Requirements: Linux+Cython+tensorflow+VOC2007
- 配置好环境后运行demo：

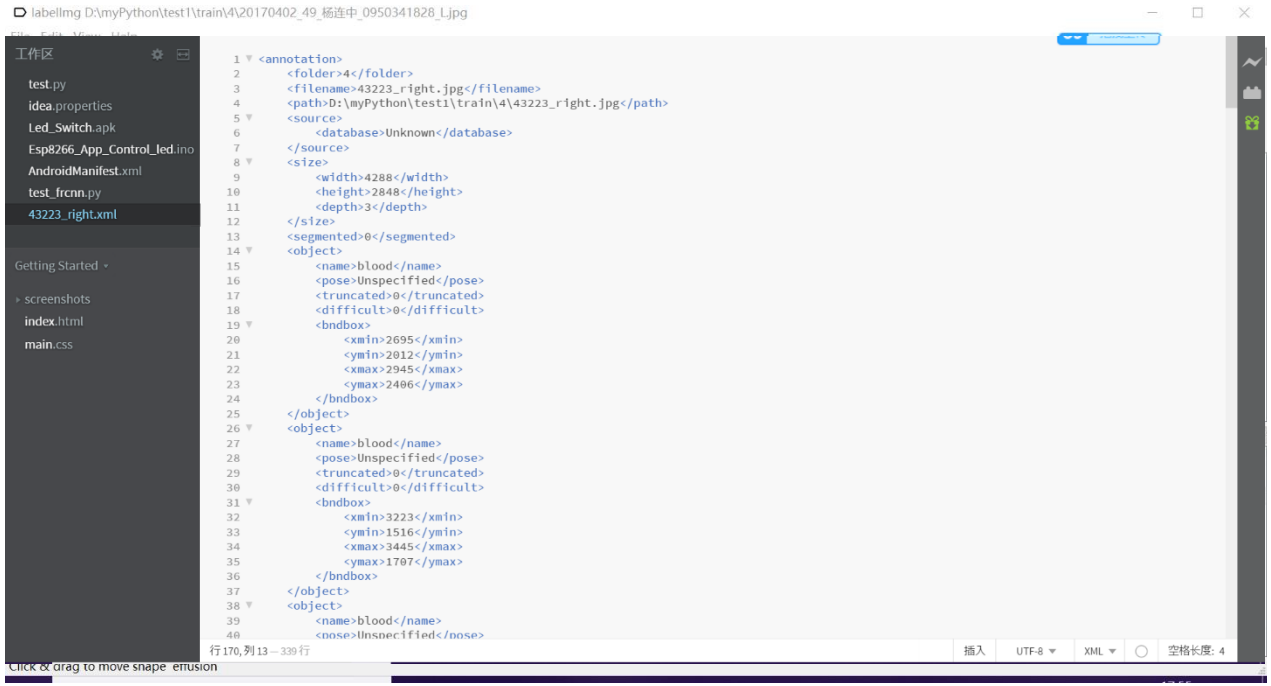
  从指定图片中识别出需要分类的物件



Approach-2

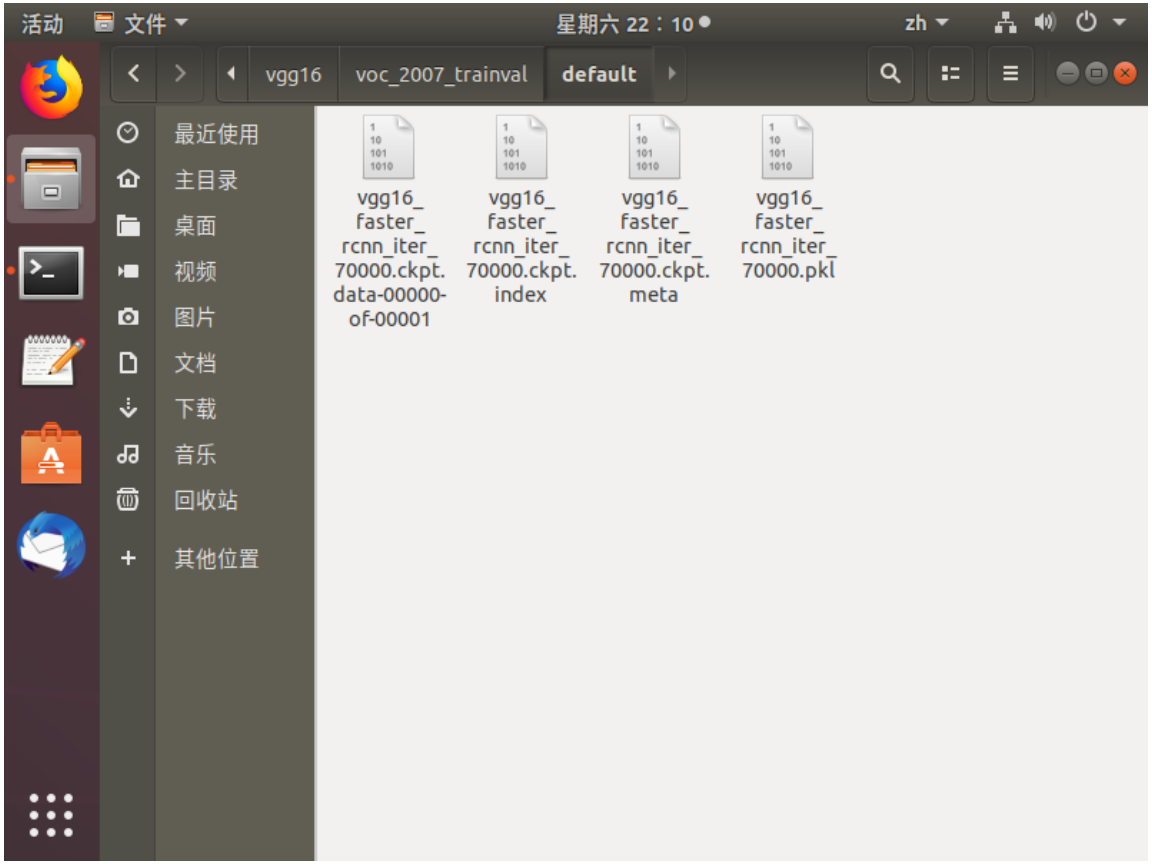
2. 制作自己的数据集

- 有三种需要识别的病理区域：微血管瘤、视网膜出血、硬性渗出
- 利用labelImg进行标注，自动得到xml文件



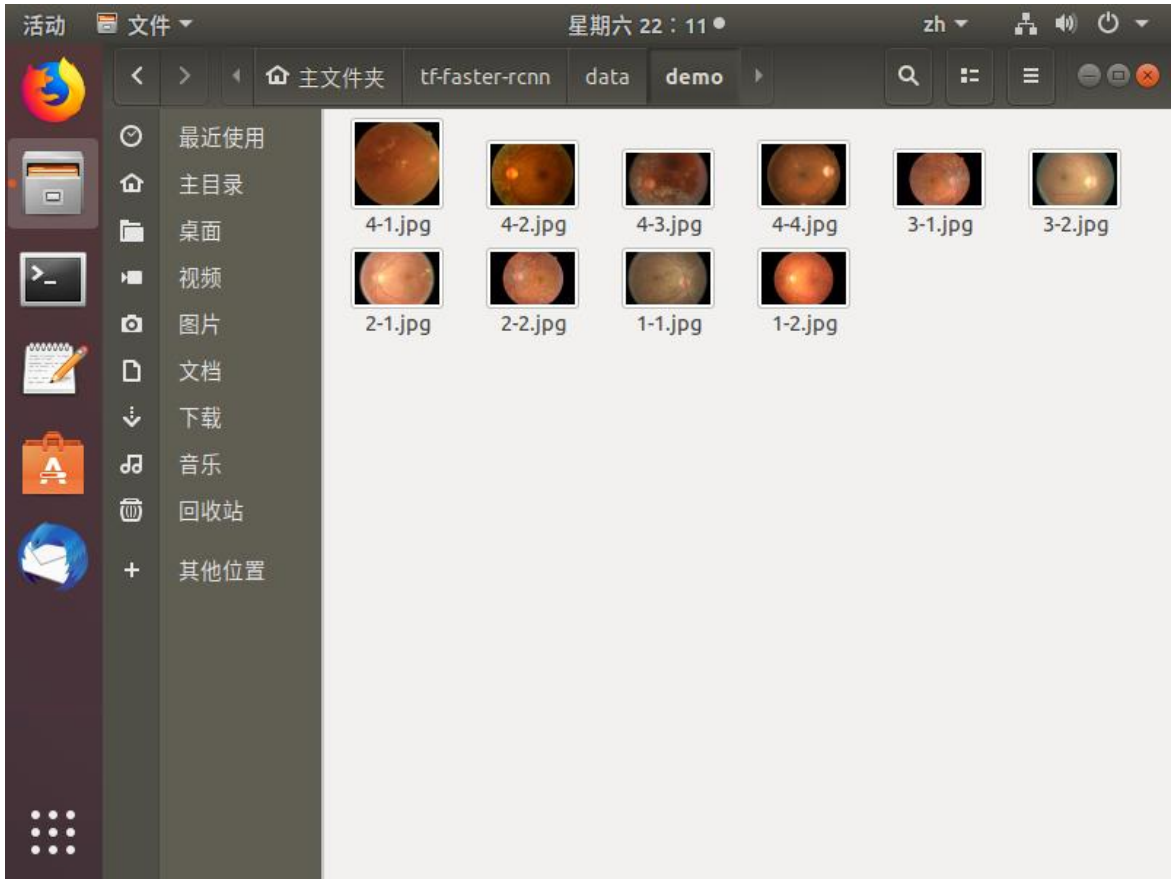
Approach-2

3. 利用VOC2007+vgg16迁移模型进行训练



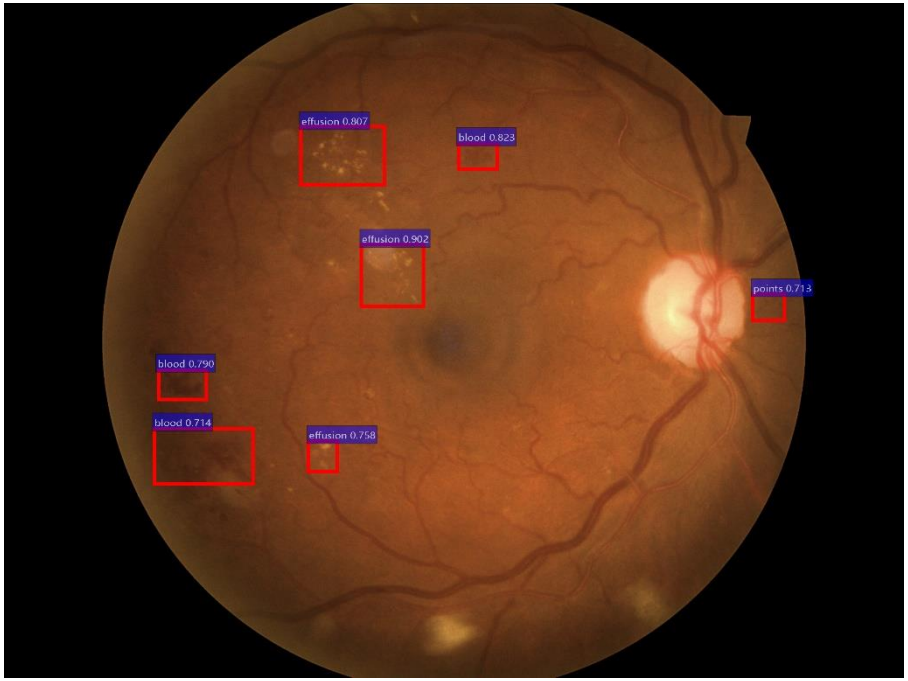
Approach-2

4. 利用训练所得模型预测图片



Outcome

- 整体准确度较好，召回率和准确度在70%左右。
- 可以观察到神经网络对硬性渗出的预测较好，而对其他两种病理表现的预测容易出现过度预测或预测不完全的情况。这可能是前期手动标记不够准确导致的。



Approach-2

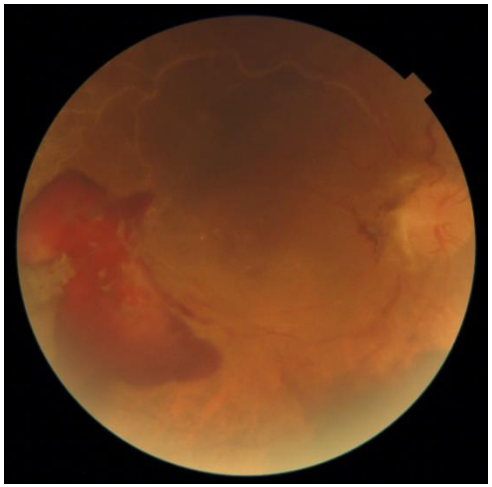
5. 可改进及优化的部分

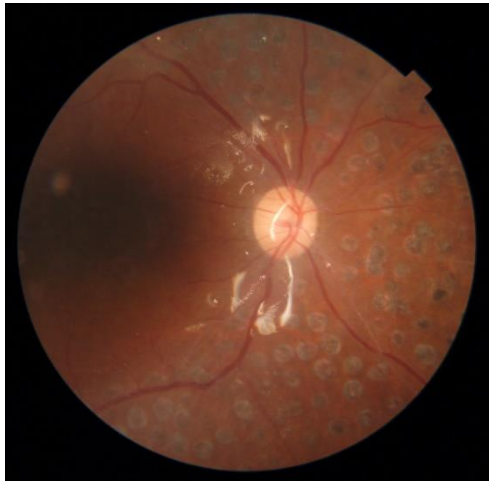
- 由于使用的是高速CPU而非GPU训练模型，图片又较大，只能使用较小的batch_size，迭代次数也较少，模型并没有完全优化
- 手动标注存在一定误差
- 只标注了等级为4的100张图片（考虑到严重病患的病理切片更具有代表性），导致训练集可能较小。后期考虑进行更大量更详细的标记。
- 由于faster-rcnn训练得到的是图片上的特定区域，没有根据区域大小及数目对图片进行分类，后期可以考虑修改tensorflow的faster-rcnn代码实现完全分类。

Approach-2

5. 可改进及优化的部分

- 在标记中发现许多重症患者的三种病理表现不明显，而视网膜出血及玻璃体出血、新生血管、纤维血管增殖膜、棉绒斑等问题更为严重，可能导致这一方法的预测效果不佳。后期可以对这些问题进一步标注。





收获

- 了解并实践了图像处理的相关方法；
- 使用Keras搭建神经网络，对神经网络的架构有了初步了解；
- 在架构的过程中学习了相关优化方法，了解Adam、Dropout等特殊函数；
- 对faster-rcnn有了初步了解，利用这一网络迁移训练了自己的模型及数据，得到了较满意的结果。

谢谢！