

# 32组团队作业——回形针计数

---

## 团队成员及分工

---

邓皓元：数据预处理

丁盛为：模型选择与设计

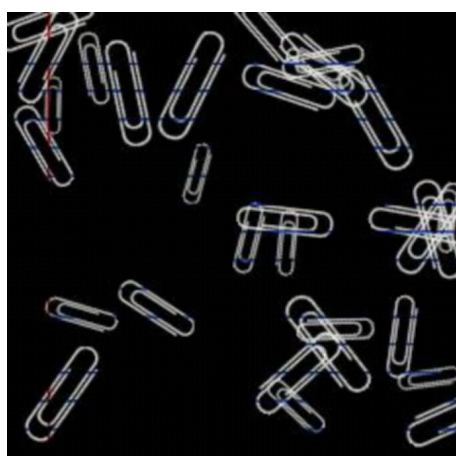
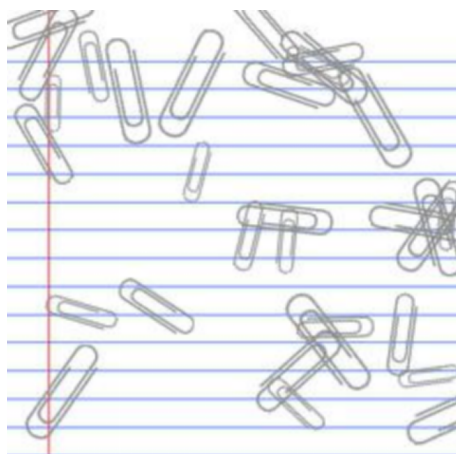
季颢天：模型的调试与运行

## 数据的预处理

---

### 背景的去除

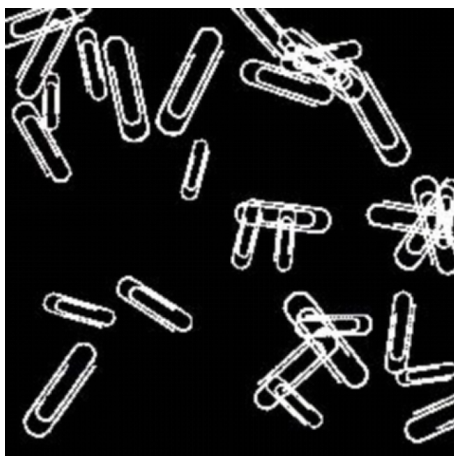
对于本次的数据集，我们发现对于所有的回形针图片都有一个相同的背景，所以可以将回形针数量为0的图片作为背景，与存在回形针的图片相减得到没有背景的数据集。首先读取照片转化为RGB三通道，再通过ImageChops.subtract方法进行做差。



原始图片与去除背景后的图片的对比.jpg

## 数据增强

由上图看出虽然去除了大部分背景，但还是存在红蓝线，考虑到可能是由于回形针的遮挡导致。通过红色和蓝色在RGB通道中的数值特点去除。对于红线考虑R通道 $>100$ ，对于蓝线考虑B通道 $>100$ 。对于RGB三通道都 $<60$ 的像素点更新为(255,255,255)，从而实现数据增强。



增强后的图片.jpg

## 扩充数据集

原先考虑通过随机数将随机图片进行随机角度的旋转。其中图片有2/3的概率被选中，旋转角度限定在120-170以及190-240之间，从而实现数据集的扩充。之后发现可能会漏掉角上的回形针，于是随机旋转角度改为90、180、270。最终数据集为newpics目录下经过去除背景和增强的图片和rotate目录下经过旋转的图片，共4万张。

## 网络模型的选择、调试与结果

本次作业中，我们采用了3种方法实现任务：CNN+全连接层，迁移学习vgg-16，迁移学习effective net

CNN+全连接层的设计中，我们使用的是CNN + BatchNorm2d + Relu的模型设计。不采用池化层是因为本次任务中对回形针的计数面对的困难是小尺度+重叠，我们认为池化也许会导致重叠时的信息丢失，因此采用了如下设计：

```
self.ch = 16 # 超参数，可以自己调整
self.cnn = nn.Sequential(
    nn.Conv2d(3, self.ch, 4, 2, 1), # 4*128*128 -> 16*64*64
    nn.BatchNorm2d(self.ch),
    nn.ReLU(),
    nn.Conv2d(self.ch, 2 * self.ch, 4, 2, 1), # 16*64*64 -> 32*32*32
    nn.BatchNorm2d(2 * self.ch),
    nn.ReLU(),
    nn.Conv2d(2 * self.ch, 2 * self.ch, 4, 2, 1), # 32*32*32 -> 32*16*16
    nn.BatchNorm2d(2 * self.ch),
    nn.ReLU(),
    nn.Conv2d(2 * self.ch, self.ch, 4, 2, 1), # 32*16*16 -> 16*8*8 此时感受野为
    67*67
    nn.BatchNorm2d(self.ch),
    nn.ReLU(),
    nn.Flatten(),
```

```
nn.Linear(1024, 256),
nn.ReLU(),
nn.Linear(256, 1)
)
```

调参时我们对卷积核的个数和大小进行过一些调整，但最终运行后的结果表现一般，最好的结果为2.5左右。我们认为这是因为CNN特征提取的能力不够强，因此尝试使用VGG16来进行迁移学习。选择VGG16是因为其在ImageNet图像分类与定位挑战赛取得了极好的成绩，且其网络的特征提取能力很强。

我们使用了VGG-16的特征提取部分，然后重新设计了其分类部分，最后的网络结构如下：

```
self.VGG.classifier = torch.nn.Sequential(
    nn.Linear(25088, 512),
    nn.ReLU(),
    nn.Dropout(0.5),
    nn.Linear(512, 128),
    nn.ReLU(),
    nn.Linear(128, 1)
)
```

这是没有强化数据时的情况：LOSS几乎只能降到平均1.5左右。

最后上交的结果比较一般，我们认为这是因为VGG模型虽然可以很好的适用于分类和定位，但是不是很适用于本次的任务。

根据听其他组的分享与自己查找资料，我们也尝试使用了EfficientNet，该模型拥有更高的性能。我们调用了经过EfficientNet-b5模型，同时在最后做了和VGG类似的处理，加入了全连接层。

一开始训练发现loss下降得很慢，通过对学习率进行调整，让loss达到了一个较低的水平，20多轮的训练效果已经能显著超过我们自己建立的网络模型，单全连接层MSE可以达到1.56。进一步的改进由于时间问题没有训练完成，比较遗憾。

## 最终成绩截图

6	TEAM_32	10	12/18/22	1.562400 (6)
---	---------	----	----------	--------------