

**论文**

**基于Fast RCNN的**

**图片物体分类和定位**

|  |  |  |
| --- | --- | --- |
| **小组成员** | **：** | **刘泊、季慧、雷紫薇** |
| **班级** | **：** | **种子班1601班** |
| **指导教师** | **：** | **王兴刚** |
| **完成日期** | **：** | **2019年5月19日** |

摘 要

从二十世纪中叶起，非结构化数据就以一个很快地速度在增长。然而，所产生的数据中有80％是非结构化的多媒体内容，未能把重点放在组织大数据的举措上，这个多媒体内容的很大一部分是图像。智能无线设备的迅速普及以及通过互联网共享图像的兴起，对这类内容的大规模增长作出了重大贡献。图像反映了人类知识、互动和对话的很大一部分。如今，图像数据中的大量知识为创造新的使用案例、应用和产品创造了极大的机会。几十年来，对图像的处理，理解、识别和定位一直是人工智能（AI）和机器学习（ML）中的一个巨大的技术挑战，但在过去的十年里，已经有了一些突破。图片比文字更具有影响力，因为它们往往更具吸引力。图像更可能被共享和转发。人们利用图像/视频来捕捉他们的特殊时刻。但是，图像已经发展成为一种交流手段。基于14年R-CNN的大获全胜，微软研究所的Ross Girshick提出了改进算法 Fast RCNN 。Fast R-CNN 在VGG16 network下训练速度比R-CNN快了9倍，测试时间更是缩短了213倍，不仅如此，mAP也达到了最高水平。本文主要沿用了Fast RCNN算法进行图片的识别和定位工作，期望能正确分类图像并对图像中的对象进行准确定位。

**关键词：**图片识别、物体定位、Fast RCNN

Abstract

From the middle of the twentieth century, unstructured data has grown at a rapid rate. However, 80% of the data generated is unstructured multimedia content, failing to focus on the initiative to organize big data. A large part of this multimedia content is images. The rapid spread of smart wireless devices and the rise of images shared over the Internet have contributed significantly to the massive growth of such content. The image reflects a large part of human knowledge, interaction and dialogue. Today, the vast amount of knowledge in image data creates tremendous opportunities for creating new use cases, applications, and products. For decades, the processing, understanding, identification and positioning of images has been a huge technical challenge in artificial intelligence (AI) and machine learning (ML), but in the past decade, there have been some breakthroughs. Pictures are more influential than words because they tend to be more attractive. Images are more likely to be shared and forwarded. People use images/video to capture their special moments. However, images have evolved into a means of communication. Based on the 14-year R-CNN win, Ross Girshick of the Microsoft Research Institute proposed the improved algorithm Fast R-CNN. Fast R-CNN is 9 times faster than R-CNN in VGG16 network, and the test time is 213 times shorter. Not only that, mAP has reached the highest level. In this paper, the Fast RCNN algorithm is used to identify and locate the image. It is expected that the image can be correctly classified and the objects in the image can be accurately located.

**Key words：**picture recognition, object location, Fast RCNN

目 录

[摘 要 I](#_Toc9202034)

[Abstract II](#_Toc9202035)

[目 录 III](#_Toc9202036)

[1 实验概述 1](#_Toc9202037)

[2 相关原理及基本流程 2](#_Toc9202038)

[2.1 实验原理 2](#_Toc9202039)

[2.1 实验指导 4](#_Toc9202039)

[2.1 实验基本流程 4](#_Toc9202039)

[3 实验过程 6](#_Toc9202040)

[3.1 数据处理 6](#_Toc9202041)

[3.2 网络结构 7](#_Toc9202042)

[3.3 测试 10](#_Toc9202043)

[3.4 测试结果分析 11](#_Toc9202044)

[4 理解和体会 15](#_Toc9202045)

[致谢 16](#_Toc9202046)

# 实验概述

本次实验我们选取的是老师在ppt上展示的图像分类与定位的题目，使用CNN进行图片物体的分类和定位，图片集中的目标种类共五种，分别是猫、鸟、乌龟、狗和蜥蜴，其中每个种类的图片各有180张。需处理的图片大小为128\*128。评价指标为一个正确的定位意味着预测的类别标签是正确的同时预测IoU>0.5，最后需报告平均正确率并展示一些定位结果。

# 相关原理及基本流程

## 实验原理

传统RCNN的基础思想是：

* 在图像中确定约1000-2000个候选框
* 对于每个候选框内图像块，使用深度网络提取特征
* 对候选框中提取出的特征，使用分类器判别是否属于一个特定类
* 对于属于某一特征的候选框，用回归器进一步调整其位置

Fast RCNN针对RCNN在训练时是multi-stage pipeline和训练的过程中很耗费时间空间的问题进行改进。它主要是将深度网络和后面的SVM分类两个阶段整合到一起，使用一个新的网络直接做分类和回归。主要做以下改进:

1、 最后一个卷积层后加了一个ROI pooling layer。ROI pooling layer首先可以将image中的ROI定位到feature map，然后是用一个单层的SPP layer将这个feature map patch池化为固定大小的feature之后再传入全连接层。

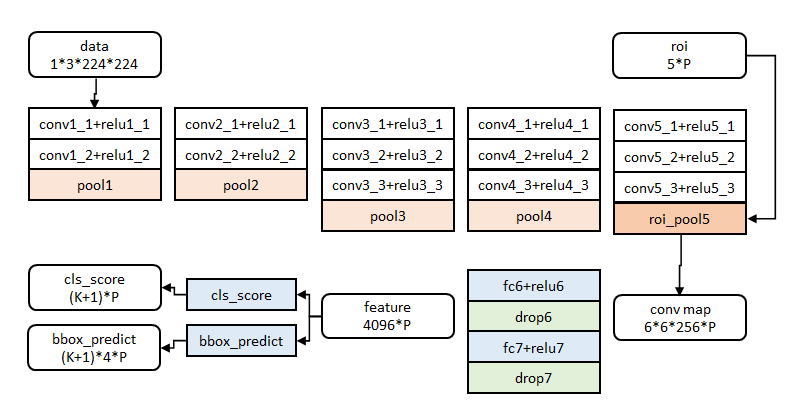
2、 损失函数使用了多任务损失函数(multi-task loss)，将边框回归直接加入到CNN网络中训练。

继2014年的RCNN之后，Ross Girshick在15年推出Fast RCNN，构思精巧，流程更为紧凑，大幅提升了目标检测的速度。

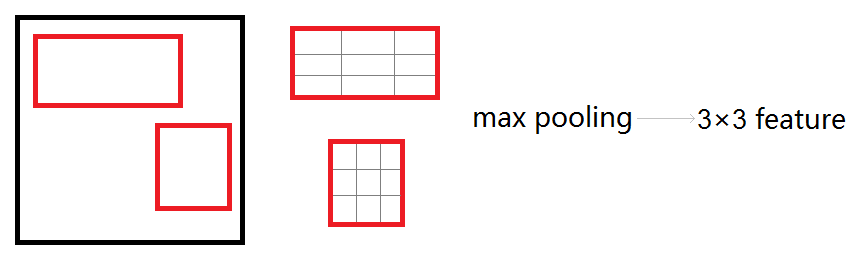
### Fast RCNN原理

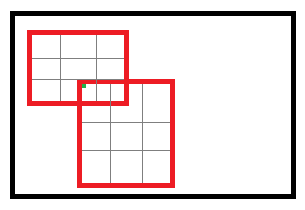
#### 特征提取网络

先构造基本结构，图像归一化直接送入网络。前五阶段是基础的conv+relu+pooling形式，在第五阶段结尾，输入P个候选区域。



Roi\_ pool层的测试，Roi\_ pool层将每个候选区域均匀分成M×N块，对每块进行max pooling。将特征图上大小不一的候选区域转变为大小统一的数据，送入下一层。





#### 网络参数训练

先进行参数初始化，网络除去末尾部分如下图，在ImageNet上训练1000类分类器。结果参数作为相应层的初始化参数。其余参数随机初始化。接下来对数据分层，在调优训练时，每一个mini-batch中首先加入N张完整图片，而后加入从N张图片中选取的R个候选框。这R个候选框可以复用N张图片前5个阶段的网络特征。最后构成训练数据。

#### 分类与位置调整

第五阶段的特征输入到两个并行的全连层中，cls\_score层用于分类，输出K+1维数组p，表示属于K类和背景的概率。 bbox\_prdict层用于调整候选区域位置，输出4\*K维数组t，表示分别属于K类时，应该平移缩放的参数。代价函数由两部分构成，loss\_cls层评估分类代价，loss\_bbox评估检测框定位代价。总代价为两者加权和。最后用全连接层提速。

## 实验指导

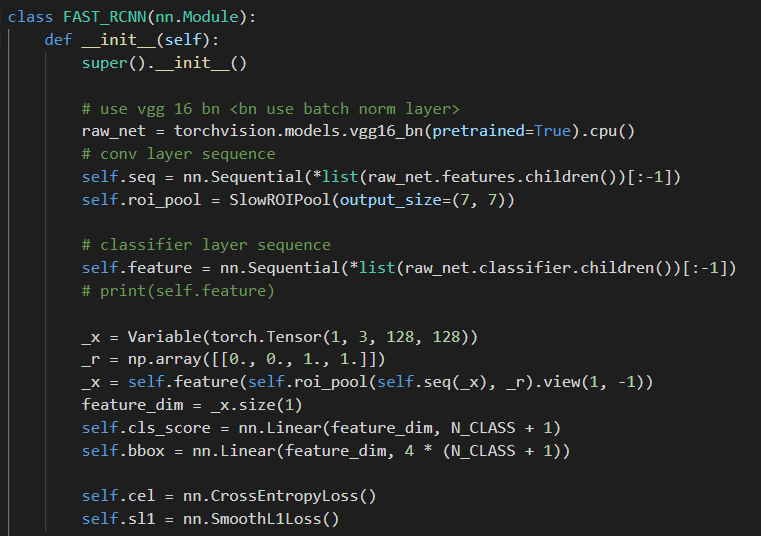
在Fast RCNN原理指导下，我们针对数据集给出了相应的指导

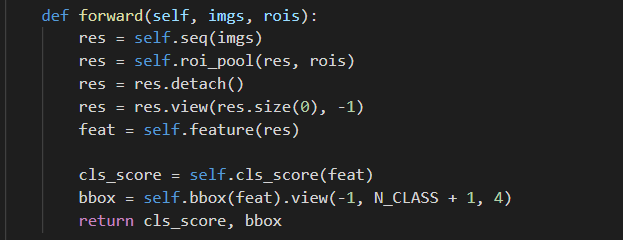
* 输入：3 x 128 x 128 RGB图像
* 卷积层：使用卷积层提取图像的特征
* ROI层：ROI层将搜索框框住的部分应用最大池，转化成统一的数据发送到下一层
* 全链接层：通过全链接层分类
* cls\_score：得到图像的分类信息
* bbox\_predict：得到图像的位置信息

## 实验基本流程

使用Fast RCNN作为框架模型。Fast RCNN模型首先使用VGG16作为基础模型，然后对卷积层的最后一层修改，并加入ROI层，ROI层将location数据与图片数据相结合，得到感兴趣的区域。然后通过线性分类层，在最后输出加上两个并行的线性分类层，分别得到5个label的概率信息和4 x 5的location信息。使用CrossEntropyLoss计算得分的Loss，使用SmoothL1Loss计算位置信息的Loss，将两个Loss按比例相加得到最后的Loss

代码实现如下





# 实验过程

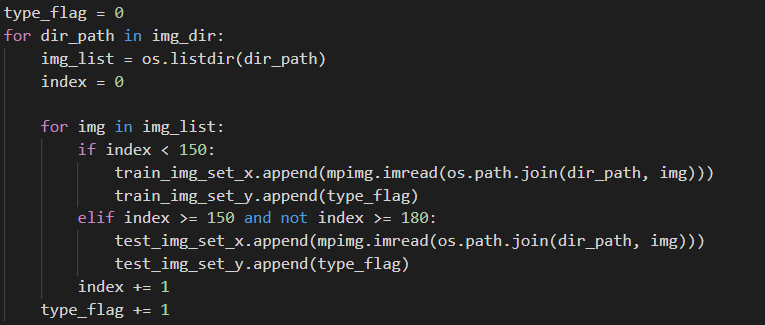
## 数据处理

给定数据是一些图片和对应的location.txt文件，读取图片并存储到变量中，同时读取location文件，将读取到的字符分割并将后四个位置存储到对应的变量中，然后将list转换成numpy数组，

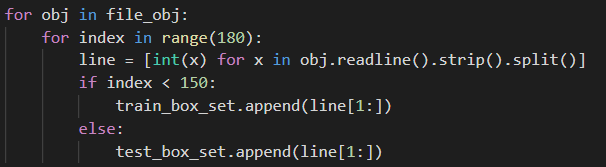
此时一共读取了750张图片作为训练集，150张图片作为测试集，shape为（750,128,128,3）和（150,128,128,3）；对应的750组location训练数据，150组测试数据，shape为（750,4）和（150，4），由于图片的shape与需要的shape有出入，所以使用transpose做维度变换，将图片测训练集和测试集的shape转换为（750,3,128,128）和（150,3,128,128）。

最后将测试集，训练集对应的图片数据，label数据和location数据用同样的方式打乱，得到最终的训练数据和测试数据/

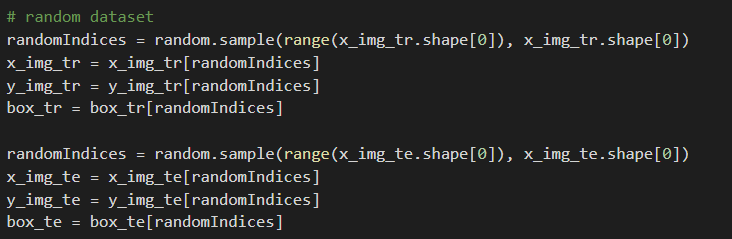
加载图片：



加载位置数据：



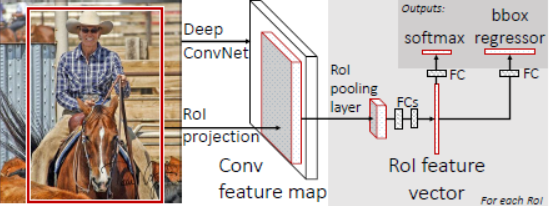
对数据重新排序：



## 网络结构

输入的是3 x 128 x 128 RGB固定大小的图像，经过5个卷积层+2个降采样层（分别跟在第一和第二个卷积层后面），进入ROIPooling层（其输入是conv5层的输出和region proposal，region proposal个数大约为2000个），再经过两个output都为4096维的全连接层，分别经过output各为5和20维的全连接层（并列的，前者是分类输出，后者是回归输出），最后接上两个损失层（分类是softmax，回归是smoothL1）。

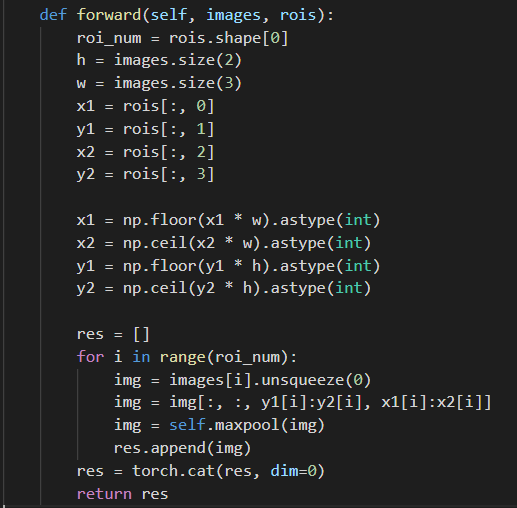
Fast RCNN模型的流程图如下：



### ROIPooling

由于region proposal的尺度各不相同，而期望提取出来的特征向量维度相同，因此需要某种特殊的技术来做保证。ROIPooling的提出便是为了解决这一问题的。其思路如下：

1. 将region proposal划分为H×WH×W大小的网格
2. 对每一个网格做MaxPooling（即每一个网格对应一个输出值）
3. 将所有输出值组合起来便形成固定大小为H×WH×W的feature map
4. 本次实验简化流程，采用region proposal然后做MaxPooling直接输出，得到固定大小的图像。代码如下图所示。

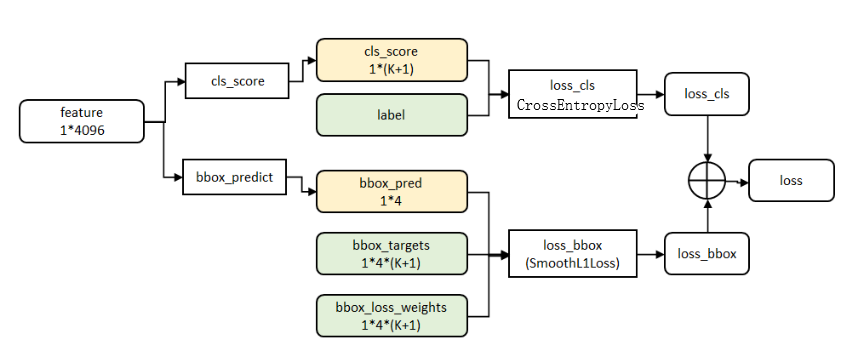


### 训练样本

训练过程中每个mini-batch包含50张图像和50个对应的region proposal（即ROI，1个ROI/张）。将图片数据和label数据和位置数据转换为Tensor之后，输入到网络中。

### 特征提取结构

1. cls\_score层用于分类，输出K+1维数组p，表示属于K类和背景的概率。
2. bbox\_prdict层用于调整候选区域位置，输出4\*K维数组t，表示分别属于K类时，应该平移缩放的参数。

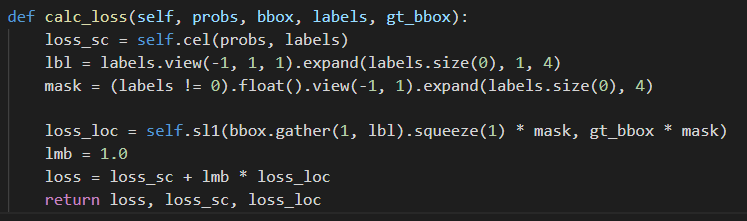


### 损失函数

使用CrossEntropyLoss计算分类的Loss，使用SmoothL1Loss计算候选区位置的Loss。

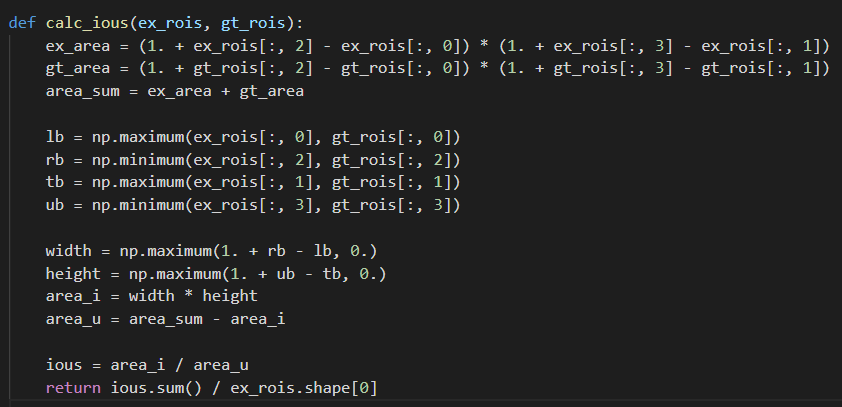
交叉熵损失CrossEntropyLoss测量分类模型的性能，该分类模型的输出是0到1之间的概率值。交叉熵损失随着预测概率偏离实际标签而增加。

Smooth L1损失用于对某些物体检测系统进行盒式回归（SSD，快速/快速RCNN），这种损失对异常值的敏感性低于其他回归损失。

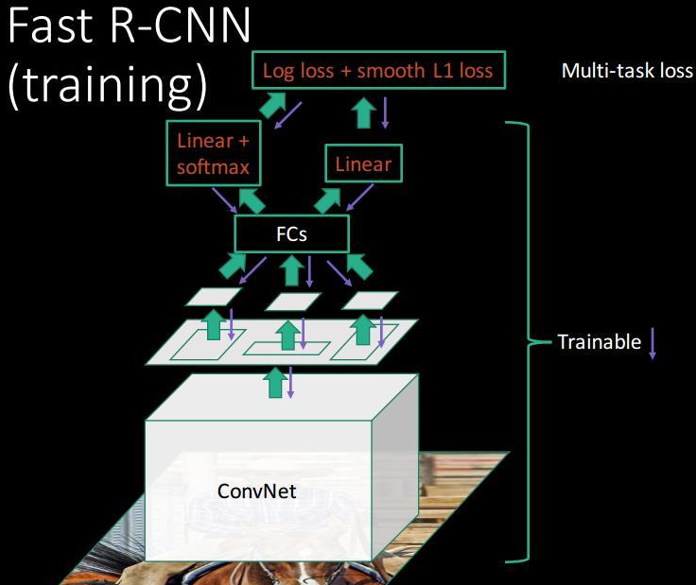


### IOU计算

IOU计算如下，首先计算实际的框的面积和标出的框的面积，然后计算出重叠的面积，然后通过 重叠面积/（总面积 – 重叠面积）计算出IOU大小



### 训练整体框架



## 测试

测试与训练类似，将150 x 3 x 128 x 128大小的图片输入到训练好的网络中，得到5种类的分数，和4 x 5的位置信息。

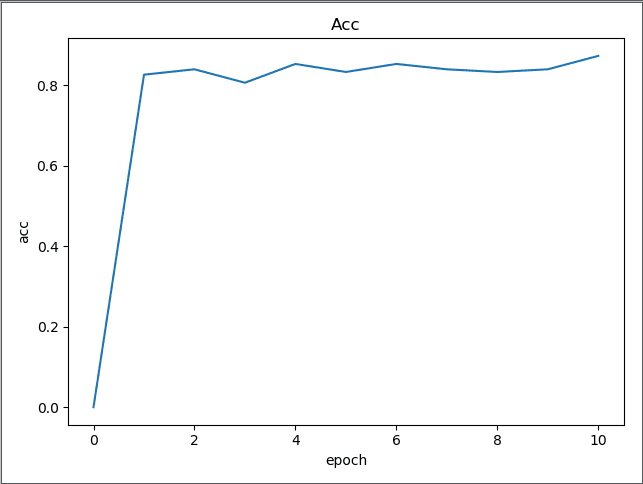
将得到的分数，选择一个最大的概率的坐标就是该图片的种类，然后与label数据进行对比，得到正确率信息。

将得到的位置信息，由上面得到的种类信息，选择一个位置信息，然后与正确的位置信息对比计算得到IOU

## 测试结果分析

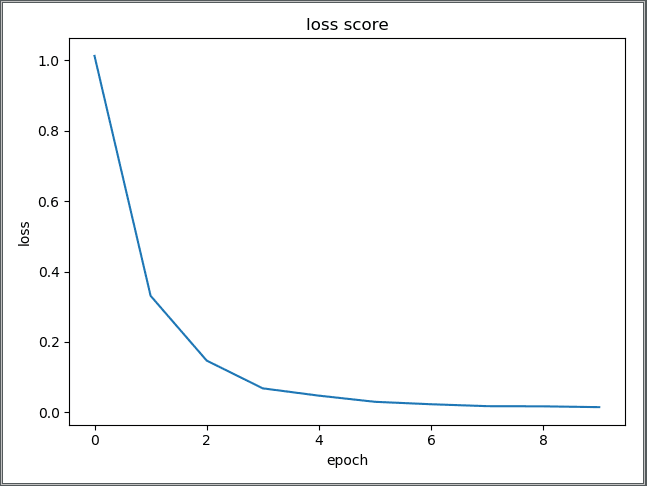
### 正确率

由于我们使用的基础网络VGG是经过预训练的，所以一开始正确率就比较高，都在0.8以上，如下图所示。

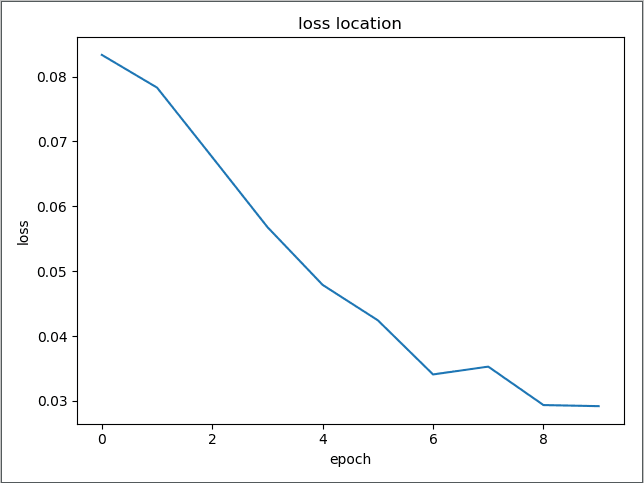


### LOSS

种类的分类loss随着epoch增大变化情况如下图所示。

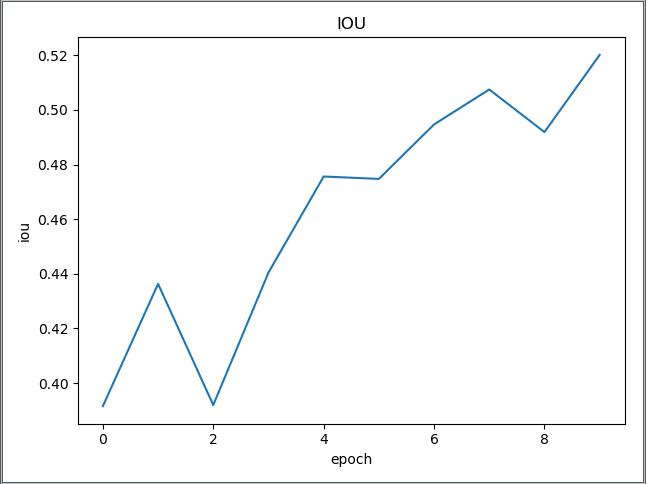


位置Loss随着训练的epoch增大变化情况如下图所示。

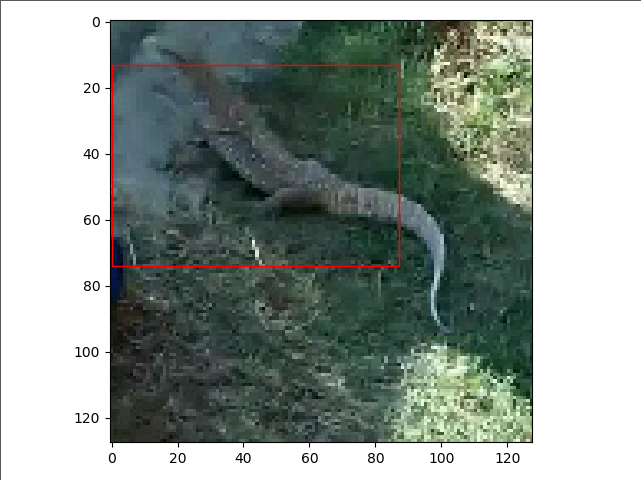
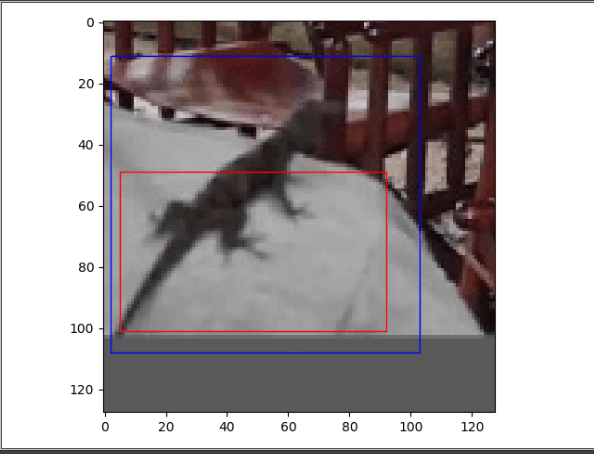


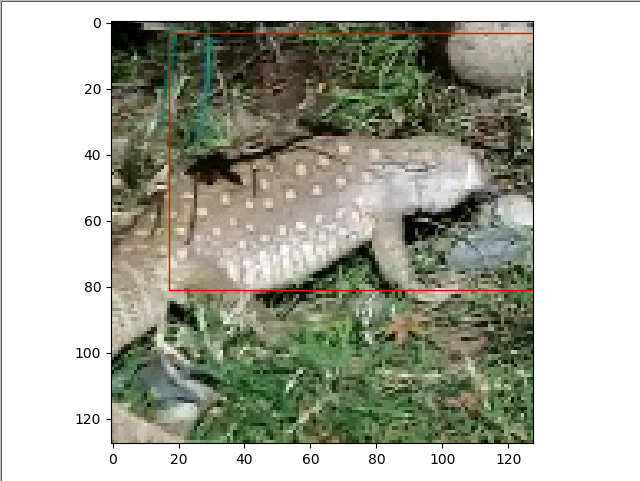
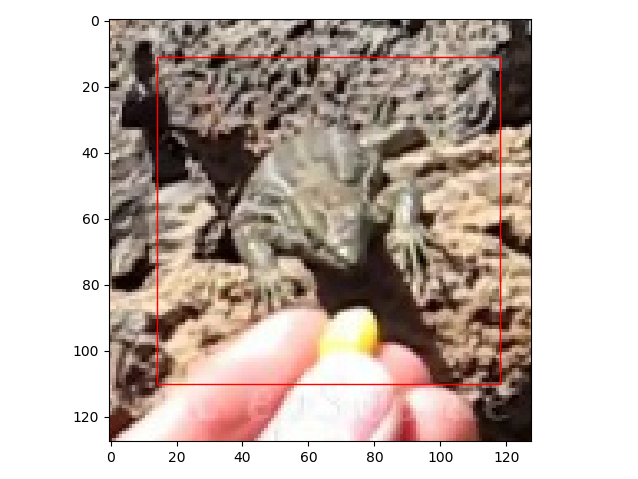
### IOU

随着epoch的增大，IOU变化情况如下图所示，最终为0.52 > 0.5



抽取部分图片，查看标注的框的信息，如下图所示：





# 理解和体会

在本次实验中，首先遇到了网络选择的问题，由于FAST-RCNN出现比较早资料可能比较多，而且训练速度和训练结果也比较可观，Fast-RCNN借鉴了SPP的思路，在不用图像金字塔池化的方式下，提出简化版的ROI池化层。通过加入候选框映射功能，使网络能够反向传播。YOLO不再采用two-stage的这种思路，将图像分类和位置检测完全集成在一个网络里，将边框位置设计称可以回归的参数，直接由网络回归得出。速度极大的得到提升，但是检测精度跟faster-RCNN相比，降低了不少。所以就选择了FAST-RCNN作为我们的基础网络，后来发现关于FAST-RCNN网络的资料很少，研究这个网络的人也比较少，可能是因为FASTER-RCNN的出现比较快然后就把FAST-RCNN给淹没了吧。然后我们根据论文和前人做的实验，经过修改以适应我们的训练模型，得到了FAST-RCNN的基本网络结构，将数据集加载并转换成一定的可输入的格式，最后将数据输入到网络并对训练的网络进行测试。

这个实验最关键的部分在于ROI部分，其余的网络模型例如FASTER-RCNN等与FAST-RCNN的区别就在于ROI部分，FAST-RCNN的ROI层中一张图片中是有很多的区域可以选择的，但是我们的数据集每张图片只有一个标注区域，所以在这部分对原模型进行了修改，每个图片只提取出一个ROI然后经过最大池化后输出到下一层中。

经过实验我们对图像识别这个领域有了一定的初步理解，从服务器环境的构建到处理数据，从理论学习和动手实验，真切感受到了算法的一点一点实现过程，收获良多。

致谢

本次实验过程中，我们得到了很多在帮助。有很多机器学习的老师在网上公布了他们的学习资料和笔记，而很多框架也已经开发完善，正是因为高度活跃的开源社区，我们的学习才会这样便利，感谢他们。同时，也很感谢小组成员在项目压力十分大的情况下仍然勇于担当，共同帮助，一起交流学习，最后完成实验，感谢！