**神经网络与深度学习期末作业汇报（选题一）**

王祯祎 20210840015

蒋 婷 20210980046

李佳睿 20210980076

林大渝 20210840010

**综述**

深度学习的卷积神经网络模型在图像处理上用处广泛，但是如果数据集中的图像数量很少，或者模型测试时的数据与训练时的数据相距甚远，则准确率可能不会很高。数据增强是解决此类问题的手段之一。数据增强是一种在训练过程中通过预先处理给定数据来增加数据多样性的方法，具有以下作用：

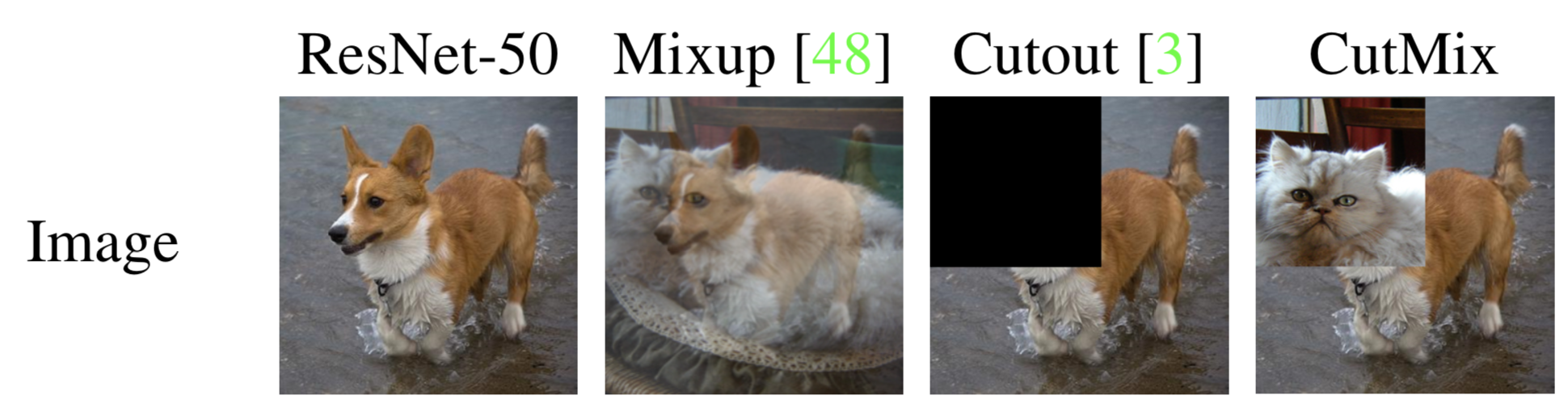
（1）避免过拟合。当数据集具有某种明显的特征，例如数据集中图片基本在同一个场景中拍摄，使用Cutout方法和风格迁移变化等相关方法可避免模型学到跟目标无关的信息。

（2）提升模型鲁棒性，降低模型对图像的敏感度。当训练数据都属于比较理想的状态，碰到一些特殊情况，如遮挡，亮度，模糊等情况容易识别错误，对训练数据加上噪声，掩码等方法可提升模型鲁棒性。

（3）增加训练数据，提高模型泛化能力。

（4）避免样本不均衡。

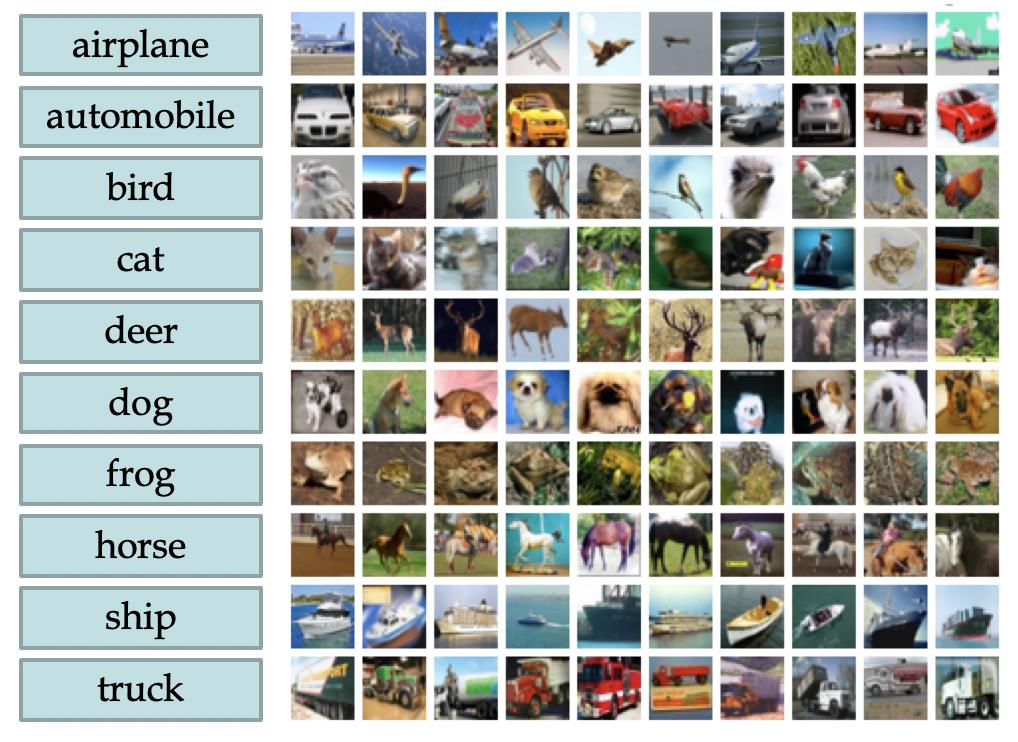
本次报告基于Cifar-10数据集和 ResNet18 神经网络比较了数据增强方法 Mixup、Cutout、CutMix与baseline的差别。



四种数据增强方法图

**Cifar-10数据集介绍**

Cifar-10是自然彩色图像数据集，被广泛用于评估深度神经网络的识别能力。 该数据集由 60000 张 32×32 像素的小图像组成，其中训练集50000，测试集10000。图像分为十个类别：airplane、automobile、bird、cat、deer、dog、frog、horse、ship、truck，每类 6000 张图像。以下是数据集中的分类，以及每个类中的 10 张随机图像：



Cifar-10分类图

**数据增强方法简介：**

**1、Mixup：**将不同类之间的图像进行混合，从而扩充训练数据集**。**

**2、Cutout：**对一张图像随机选取一个小正方形区域，在这个区域的像素值设置为0或其它统一的值从而实现局部遮挡。

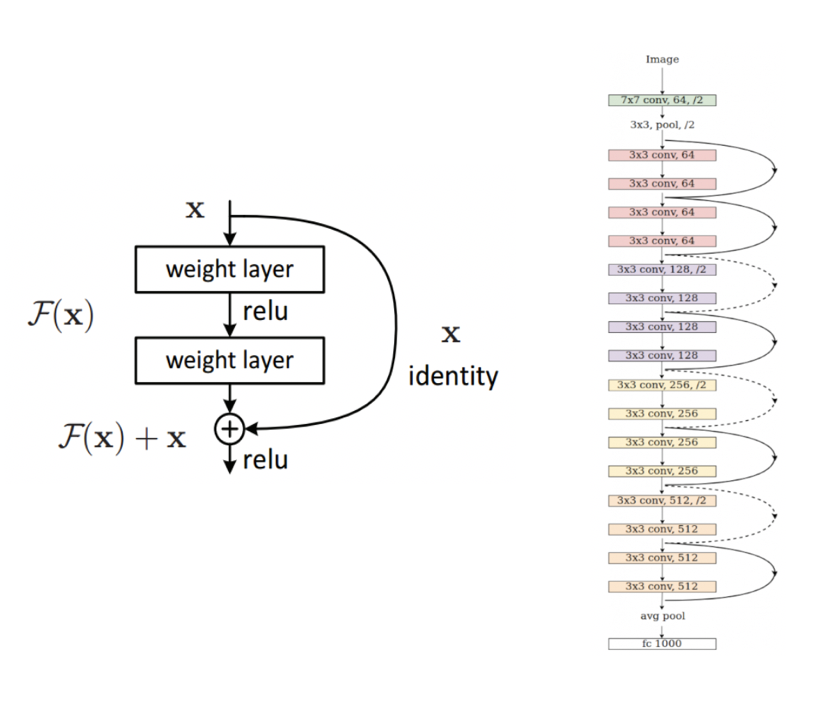
**3、CutMix：**将一部分区域cut掉，但不填充0像素，而是随机填充训练集中的其他数据的区域像素值。

**Baseline方法：ResNet-18 + 未经处理的Cifar-10数据集**

已有研究证明，向神经网络添加更多层可以使其对与图像相关的任务更加稳健。但同时，如果给网络添加太多层，那么就会产生梯度消失和退化问题，网络性能就会受到影响。因此 ResNet应运而生。ResNet 是一种在传统模型中引入残差块的模型，通过引入残差块，可以比具有多层的传统模型更有效地学习并创建高性能模型。根据残差网络的层数，ResNet 主要有 5 个不同的变体，分别是 ResNet18、ResNet34、ResNet50、ResNet101 和 ResNet152。在这个项目中，我们使用 ResNet18，它由 1 个卷积层、8 个残差块（每个残差块有 2 个卷积层）和 1 个全连接层组成。

网络结构中存在两种基础块，1种基础块是实现标注跳跃连接的部分，在通道数不变的情况下，进行的残差结构运算，如下图左边所示。另一种是带有虚线标注的跳跃连接部分，在第二种的跳跃连接结构种，通道数发生了改变，于是把它单独做成一个基础块。

下图为一个Resnet-18的示意图：



ResNet残差块和ResNet18

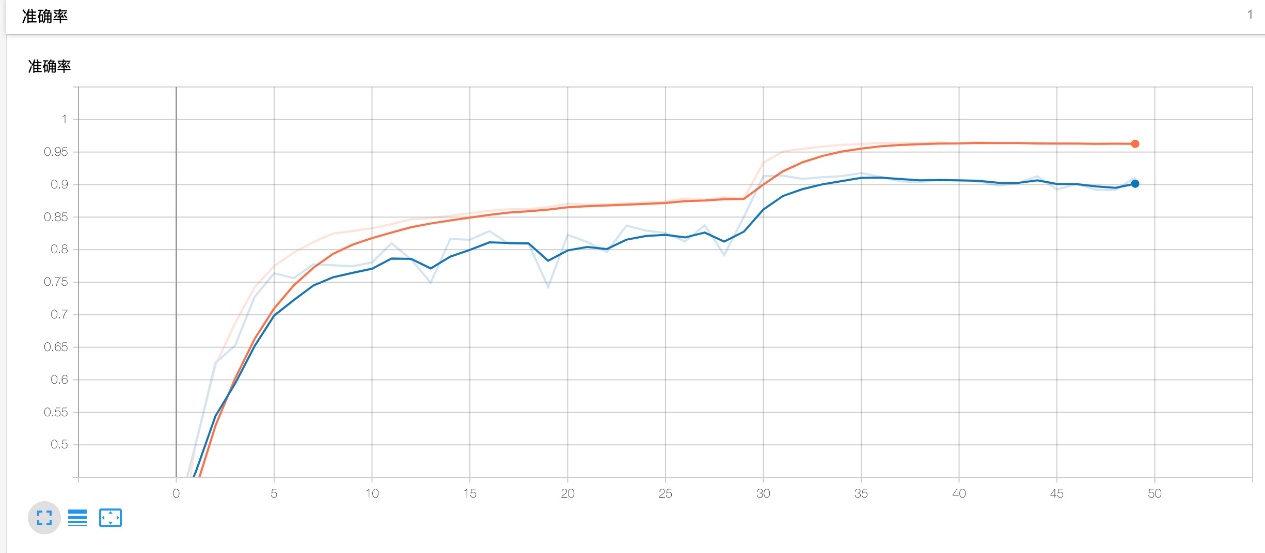
**Resnet-18参数设置：**

我们Resnet-18方法的参数设置为：batch size=128，learning rate=0.1，iteration约为39062，epoch=50，损失函数选取交叉熵，优化方式为momentm-SGD，采用mini-batch，并采用L2正则化。评价指标我们选取模型结果的精确率accuracy。

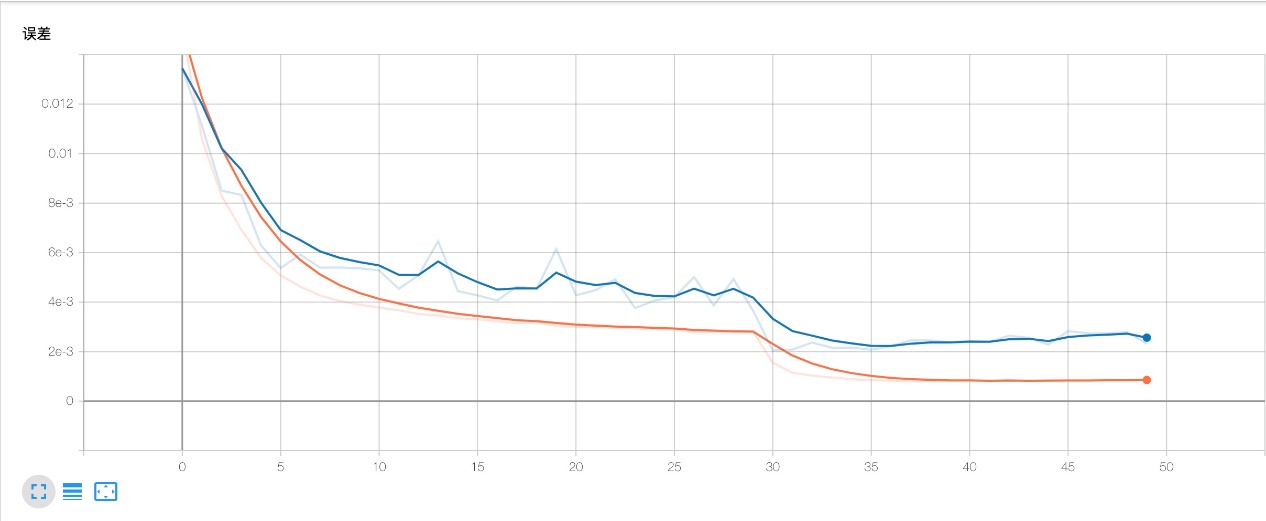
**Resnet-18运行结果：**

Resnet-18的运行结果如下图所示：

其中橙色曲线为训练集上的验证结果，蓝色曲线为测试集测试结果。每进行一轮训练后进行一次测试，从而得到迭代过程中测试精度的变化曲线。



Baseline准确率图



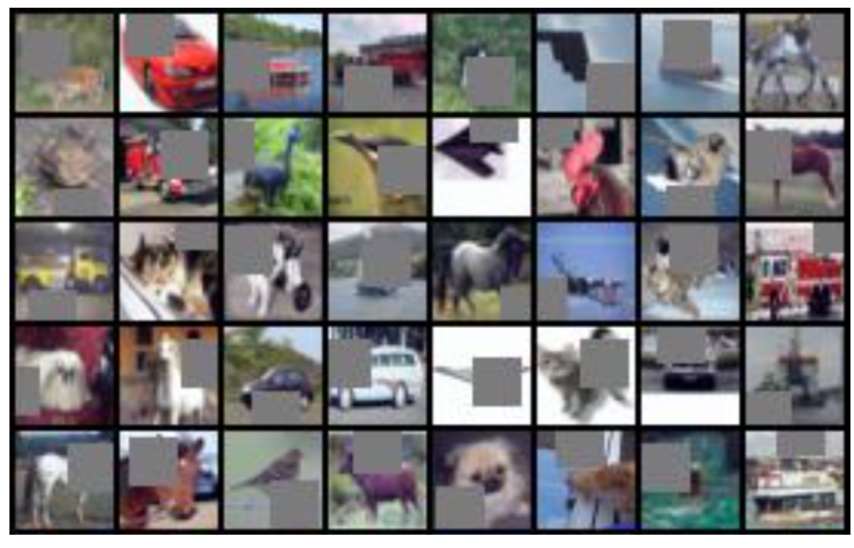
Baseline loss图

**Cutout**

Cutout是一种新的正则化方法。原理是在训练时随机把图片的一部分减掉，这样能提高模型的鲁棒性。它的来源是计算机视觉任务中经常遇到的物体遮挡问题。通过cutout生成一些类似被遮挡的物体，不仅可以让模型在遇到遮挡问题时表现更好，还能让模型在做决定时更多地考虑环境(context)。

Cutout是直接对输入的图像进行遮挡。具体来说，Cutout就是对一张图像随机选取一个小正方形区域，在这个区域的像素值设置为0或其它统一的值。

如下图，每个图片的一小部分被遮挡掉了。



Cutout应用在Cifar-10数据集上的效果图

Cutout方法有以下两点优势：

(1) 通过 Cutout 可以模拟真实场景中主体被部分遮挡时的分类场景；

(2) 可以促进模型充分利用图像中更多的内容来进行分类，防止网络只关注显著性的图像区域，从而发生过拟合。

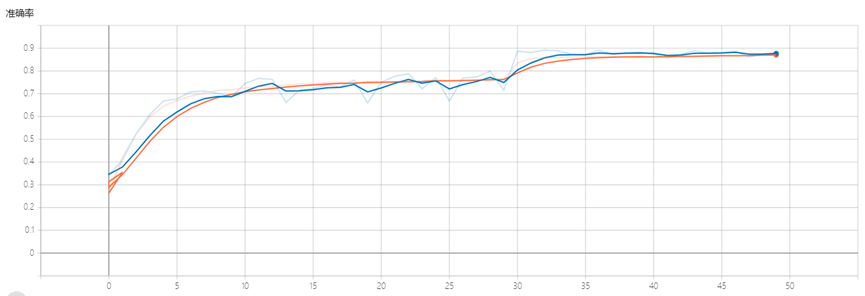
**Cutout方法参数设置：**

Cutout方法的参数设置与Resnet-18相同：batch size=128，learning rate=0.1，iteration约为39062，epoch=50，损失函数选取交叉熵，优化方式为momentm-SGD，采用mini-batch，并采用L2正则化。评价指标选取模型预测的准确率accuracy。

**Cutout方法运行结果：**

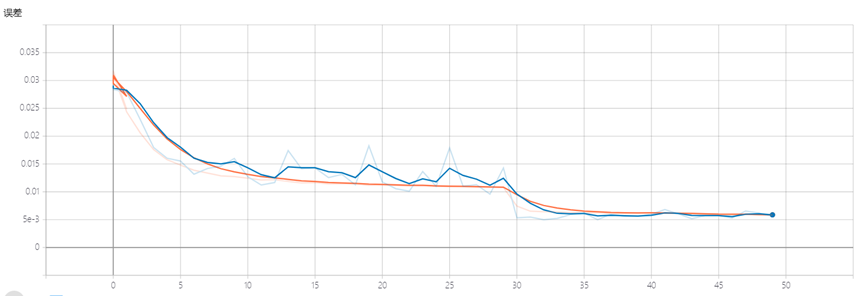
与Baseline一样，橙色曲线为训练集上的验证结果，蓝色曲线为测试集测试结果。绘制出测试精度的变化曲线。

图中准确率与loss出现了大幅的斜率变化，可能是由于越过局部最优点所致



Cutout准确率图

最终训练集的精度为0.87048，意味着约87％的图片被正确分类；测试集精度为0.8808，约88％的图片被正确分类



Cutout loss图

Training loss≈0.005818，Test loss≈0.005582，都达到了较低的水平。

Cutout后图像分类结果如下图：



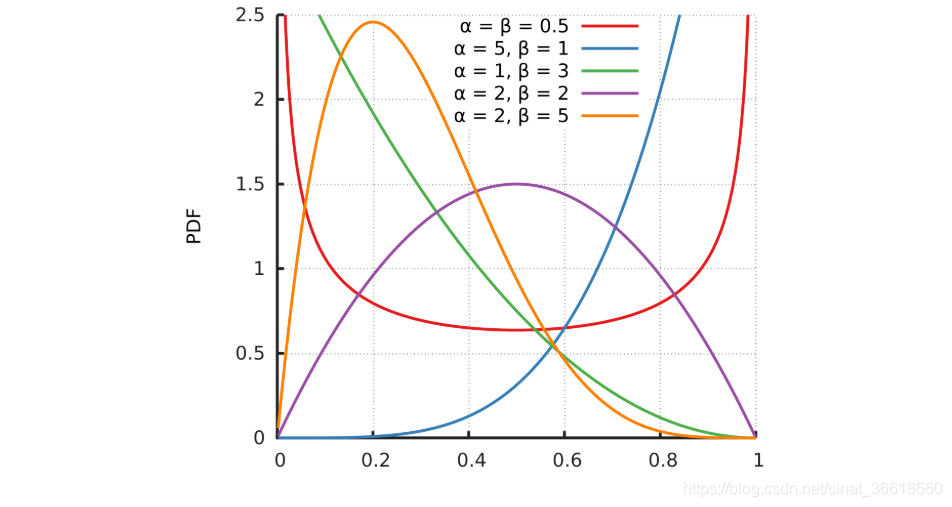
**Mixup**

**mixup原理**

假设batchx1是一个batch样本,batchy1是该batch样本对应的标签；batchx2是另一个batch样本，batchy2是该batch样本对应的标签,λ是由参数α,β的贝塔分布计算出来的混合系数，由此我们可以得到mixup原理公式为：

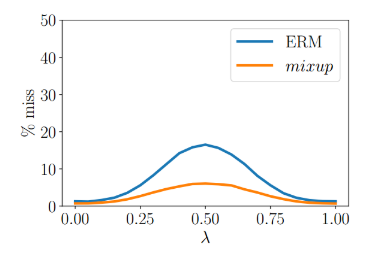
其中Beta值的是贝塔分布，mixed\_batchx是混合后的batch样本,mixed\_batchy是混合后的batch样本对应的标签。

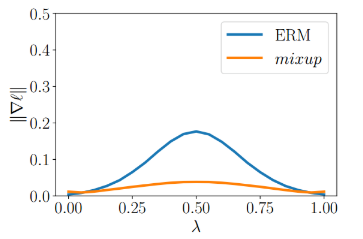
当α=β=0.5时，算法效果相对较好。Beta分布如下图：



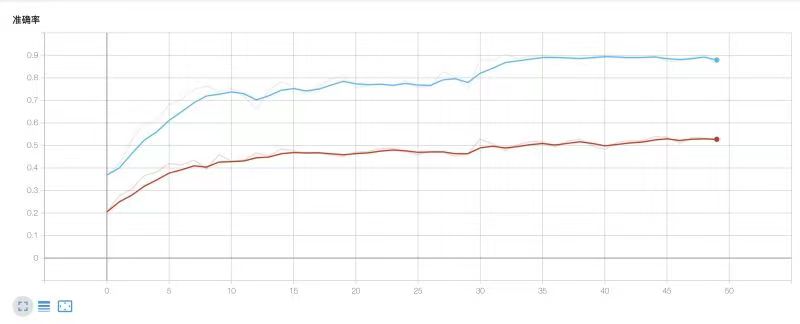
与没有太多的限制，当batch size = 1时，就是两张图片样本混合；当batch size >1时，便是两个batch图片样本两两对应混合。此外，与可以是同一批样本，也可以是不同批样本。一般在代码实现过程中，两个batch图片是同一批样本，唯一不同的是是原始batch图片样本，而是对在batch size维度进行shuffle后得到的。

mixup邻域分布可以被理解为一种数据增强方式，它令模型在处理样本和样本之间的区域时表现为线性。我们认为，这种线性建模减少了在预测训练样本以外的数据时的不适应性。从奥卡姆剃刀的原理出发，线性是一个很好的归纳偏见，因为它是最简单的可能的几种行为之一。下图显示了mixup导致决策边界从一个类到另一个类线性的转变，提供了一个更平滑的不确定性估计。第二张图显示了在CIFAR-10数据集上用mixup和ERM两个方法训练的两个神经网络模型的平均表现。两个模型有相同的结构，使用相同的训练过程，在同一个从训练数据里随机抽样而来的样本上来评估。用mixup训练的模型在预测训练数据之间的数据时更稳定。 （对离散样本空间进行连续化，提高邻域内的平滑性）

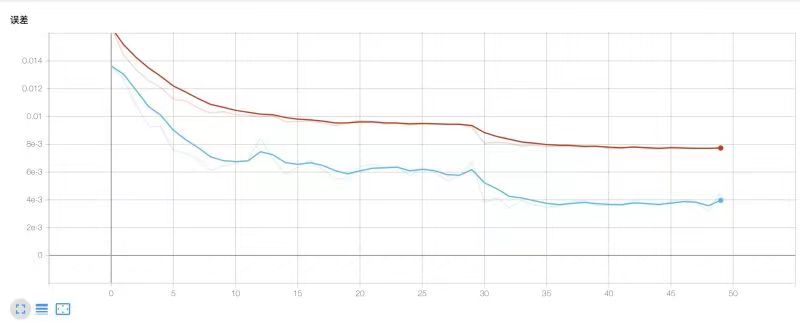




mixup预测结果：



Mixup准确率图



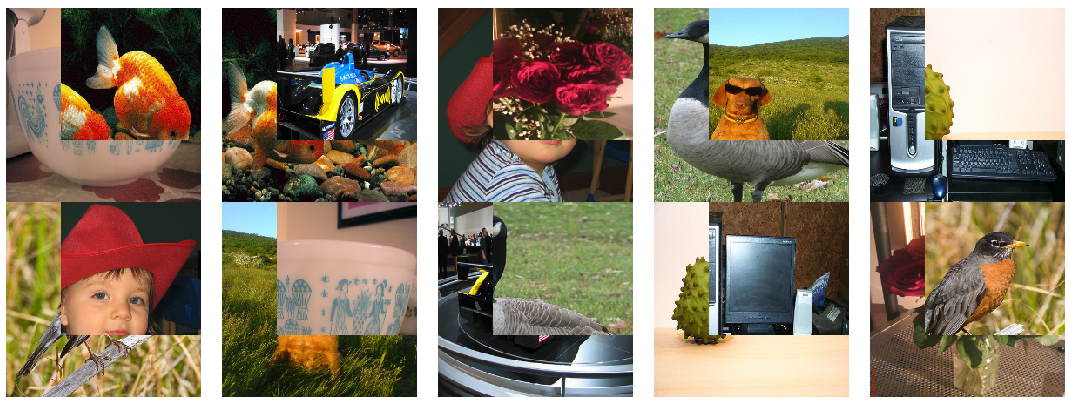
Mixup误差图

横坐标为epoch数，蓝色线代表测试集，红色线代表训练集。从图中可以看到随着epoch增加到50，测试集和训练集的误差越来越低，分别趋向于0.004和0.008。除此之外，mixup测试集样本的准确率达到了90%，而训练集样本的准确率只有50%，对此分析产生这一现象的原因。我们猜想：由于mixup的思想是两个图像的叠加，训练集中增加了经过mixup处理后变得模糊且复杂的图，因此分类难度较高，导致准确率较低；而测试集只有未经过mixup的图片，分类的难度较低，从而准确率能达到90%。

**CutMix**

**CutMix简介**

CutMix是在 ICCV 2019上提出的一种数据增强方法，与Mixup、Cutout等数据增强方法不同，CutMix是将一部分区域cut掉，但不填充0像素，而是随机填充训练集中的其他数据的区域像素值，填充结果如下所示。



CutMix具有以下的优点：

（1）在训练过程中不会出现非信息像素，从而能够提高训练效率；

（2）保留了regional dropout的优势，能够关注目标的non-discriminative parts；

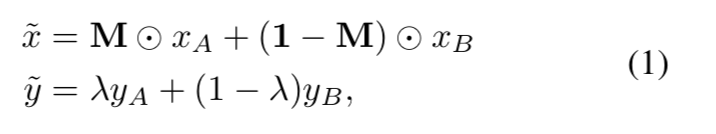
（3）通过要求模型从局部视图识别对象，对cut区域中添加其他样本的信息，能够进一步增强模型的定位能力；

（4）不会有图像混合后不自然的情形，能够提升模型分类的表现；

（5）训练和推理代价保持不变。

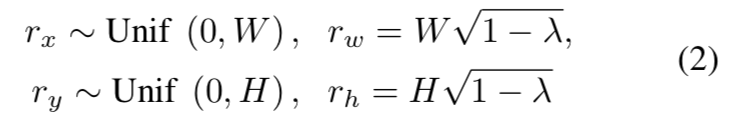
**CutMix的原理**

和是两个不同的训练样本，和是对应的标签值，CutMix需要生成的是新的训练样本和对应标签: 和, 公式如下:



其中 M ∈ {0, 1}W ×H 是为了drop掉部分区域和进行填充的二进制掩码，⊙ 是逐像素相乘，1是所有元素都为1 的二进制掩码，λ与Mixup一样属于Beta分布：λ~Beta(α, α)，令α=1则λ服从（0，1）的均匀分布。

为了对二进制掩码进行采样，首先要对剪裁区域的边界框 = (rx,ry,rw,rh) 进行采样，用来对样本和做裁剪区域的指示标定。剪裁区域的边界框采样公式如下：



保证剪裁区域的比例为, 确定好裁剪区域之后，将制掩中的裁剪区域置0，其他区域置1。就完成了掩码的采样，然后将样本A中的剪裁区域移除，将样本B中的剪裁区域进行裁剪然后填充到样本A。

**CutMix实验结果**

应用CutMix训练好的模型分别对以下10张图片进行预测，得到的预测结果及可信度分别如下图所示：



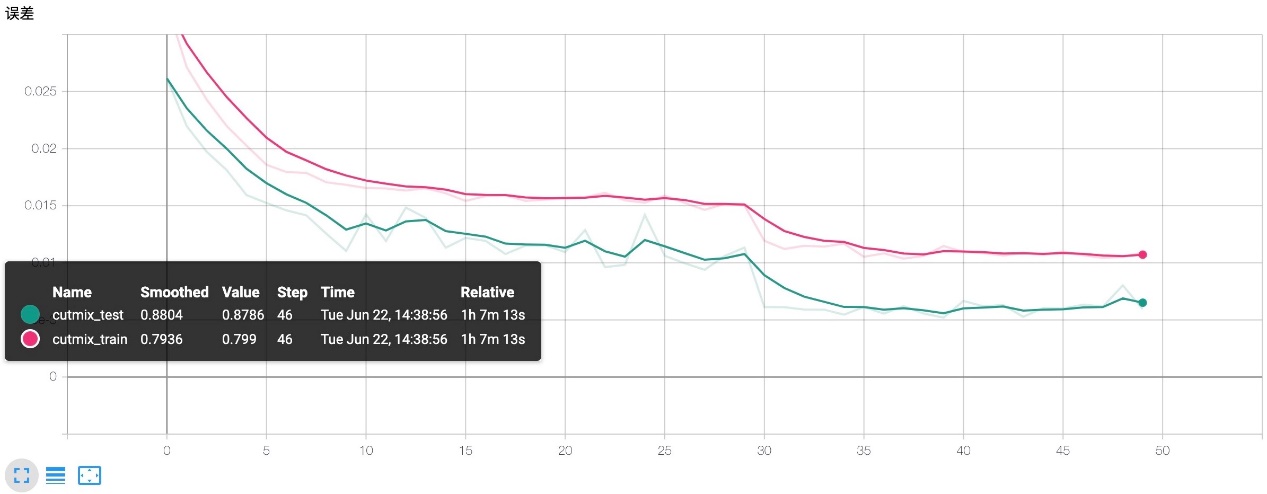
根据预测结果显示，模型对10张图像中的7张都给出了正确的预测结果，而对horse、bird和ship给出了错误分类，对应的可信度也相对较低。

模型对训练集和测试集的预测准确率分别如下图所示（其中红色曲线为训练集，绿色曲线为测试集），可以看到，测试集的准确率始终高于训练集的准确率，且随着迭代次数的增加，准确率显著上升，35次之后测试集准确率基本维持在0.88左右。



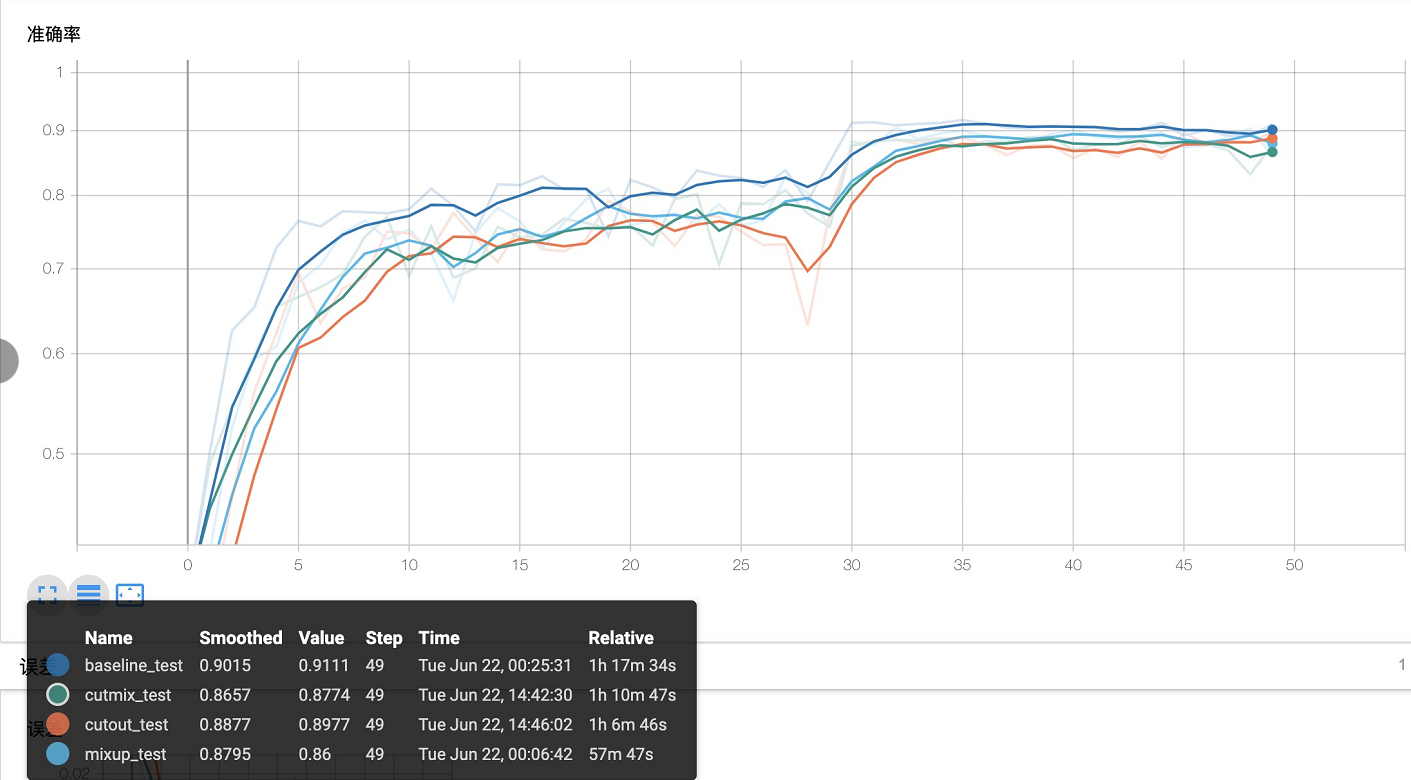
Cutmix 准确率图

在损失函数这块预测结果显示，测试集的损失函数值始终低于训练集损失函数，且随着迭代次数增加下降趋势显著，同样地，在迭代35次之后误差下降至稳定水平。



Cutmix loss图

**结果展示：**



四种方法准确率对比图



四种方法loss对比图

**小组分工**

王祯祎：撰写代码。

蒋 婷：撰写报告，主要包括综述及cutmix分析。

李佳睿：撰写报告，主要包括baseline、cutout分析。

林大渝：撰写报告，主要包括mixup分析。