# 前期工作总结报告

## 概述

为完成毕业设计（即“基于子类和深度学习的鸟类细粒度识别算法的研究与实现”），开题之后我做了相关的一些前期工作。主要包括选题的背景和意义、对深度学习的学习、对不同深度学习框架的选择、对模型的选择以及对数据集的选择等等。以下将分别对这些方面进行说明。

## 选题的背景和意义

细粒度图像识别算法是近年来计算机视觉、模式识别等领域一个非常热门的研究课题。传统的图像识别的对象都是这种狗、鸟、橘子之类，从外观就可以看出差别非常大的物体。这样的物体识别叫做粗粒度物体识别。不同于粗粒度物体识别，细粒度图像识别的目的是对属于同一基础类别的图像进行更加细致的子类划分。包括汽车、狗、花、鸟等，本次研究的对象则是鸟类。

细粒度识别的主要会面临一下几方面挑战：

1. 拍摄因素的影响。拍摄时的光照条件、图像分辨率等都可能对最终信息的采集造成较大的影响。拍摄时的光照条件可能使物体原本明显的颜色特征变得不在明显，同时图像分辨率过小也会导致原本明显的用来区分不同类别的特征不再明显，从而增加了模型训练的难度。另外如果图像分辨率过高，将可能需要修改网络结构，增加了网络的参数量，也容易造成网络的过拟合等。同时也会增加后期系统使用时的拍摄难度，因此需要在训练前确定使用多大的图像分辨率。
2. 子类别之间细微的类间差异以及较大的类内差异。同一类别的不同个体也可能有较大，类间差异度较小，背景可能会对识别结果造成干扰。所有类别之间比较相似，不同类别之间只能通过比较轻微的差异区分，进而加大了细粒度图像识别的难度。甚至有些物种可能不同性别的样本，不同年龄段的样本之间都会有较大的差异，进一步加大了网络模型训练的难度。有些物种之间可能差异极其微小，一旦由于拍摄角度，拍摄时样本的姿势等方面因素发生改变，将可能导致通过图片完全不能确定样本究竟属于哪一类别，同样也加大了神经网络模型的训练难度。
3. 数据标记困难。不同类别之间差异较小，较难以区分，标记难度较大。由于大部分模型的训练不仅需要考虑训练或测试样本所属的类别，同时还要框选除训练样本中要识别的物体所在的具体区域，否则训练的结果将会严重受到背景的影响，取得的结果可能很差，不能满足训练的期望和实际生产生活所要求的准确率。有时我们甚至还要标注训练或测试样本的各个属性，进一步增加了数据标注的工作的难度。这些标注工作需要较高的时间和经济成本，另外可能在实际生产中将所有样本都按照上面的方法进行标注很不现实。

由于以上的各方面的因素以及其他的因素，相对于普通的图像分类任务，细粒度图像分类的难度更大。准确率相对更加难以提升

同时，在诸多应用场景中，鸟类识别技术都至关重要，尤其是对于一只鸟确定该只鸟所属的类别，这也是很多其他的工作进行下去的前提，例如环境保护、濒危鸟类救援、观察气候和保护生态稳定性等场景。这些场景都对实际的生产和生活有重要的意义。因此鸟类的细粒度识别值得我们进行进一步的研究与探索。这也是我们确定该项目选题的原因。

## 对深度学习的学习

最近通过对视频教程《深度学习与PyTorch入门实战》以及书籍《动手学深度学习》的学习，对于深度学习有了一定的了解，了解了深度学习的基础以及一些常用的网络结构和函数背后的实现原理以及这些网络结构和函数的基本使用。能从零开始或者使用PyTorch来实现一些基本的网络结构。包括线性回归、softmax回归、多层感知机、卷积神经网络、循环神经网络等等。可以使用线性回归训练一个用于预测先行问题的网络模型。可以使用softmax回归加多层感知机或卷积神经网络进行MNIST手写数字数据集的简单训练与预测，并取得较高的正确率。同时，可以使用循环神经网络进行时序序列相关的问题的预测，例如将函数的x值作为时序序列的时间，预测循环函数指定时间的函数值。同时对于迁移学习、自编码器Auto-Encoders、对抗生成网络GAN等有一定的了解。可以使用迁移学习来借鉴他人训练好的网络模型来搭建自己的网络模型，同时可以使用自编码器Auto-Encoder或者对抗生成网络GAN实现以及其他的数据增强技术在样本数量较少的时候增加样本的数量，从而避免模型的过拟合。同时对于不同的优化算法、过拟合与欠拟合、交叉验证、Regularization、dropout等有基本的了解，可以进行简单的运用，从而使用交叉验证，dropout等技术来避免模型的过拟合的问题，提高模型的泛化能力，从而使得模型可以在实际的生产生活中取得较好的表现。同时也可以综合运用以上的技术搭建一个自己的神经网络模型来进行相关的研究与实验。上述所学的技术基本可以满足搭建本项目中所需的神经网络的需要。

## 对深度学习框架的选择

由于PyTorch基于动态图进行网络模型的实现，所以在搭建模型与模型的训练相对于Tensorflow来说较为容易，同时网上的中文教程与书籍等资源也较为完善，较为易于后续相关文档的查询以及定位和解决模型搭建过程中遇到的问题。另外，在学习深度学习的过程中我所使用的大部分学习资料都是基于PyTorch的，相对来说对PyTorch也更为熟悉，便于开展下一步的课题的研究和实现。因此本项目的网络模型的实现将基于PyTorch进行。

同时，本项目可能还需要使用机器学习的相关算法，由于sklearn对机器学习算法做了较为完善的封装，因此我们可能会使用sklearn从而调用相应的机器学习相关算法，从而实现相应的功能。

## 对模型的选择

最近的鸟类分类的研究主要通过寻找局部区域或者提取归一化的特征从而聚焦于位置和视角变化问题。一些研究者使用了定位鸟类或者其他动物的局部区域的方法，发现该方法可以提高分类的正确率。另一种通过提取位置归一化特征的方式也有不错的效果。除了位置和视角的变化问题之外，对于任何一个细粒度分类问题来说，一个主要挑战是怎么将具有高度的视觉相似度的不同类别区分开来。本项目将会主要关注这个问题。下面是本项目前期的基本构想。

本次的网络模型对于细粒度图像分类我们使用一种全新的实现方式。首先，我们的系统使用一个粗略的分类器（也即子类选择器），将训练或测试样本分配给一个子类中。之后，我们需要判断当前子类与测试样本之间的置信度，会将置信度与一个阈值进行比较，如果置信度足够高（高于指定的阈值），那么对于每一个被选定的子集，我们将会使用一个局部分类器对该样本进行分类。对于每个子类，我们将会为它训练一个SVM分类器，不同的子类的分类器将会拥有相同的结构以及分别训练出来的不同的参数，该分类器将会被用于区分属于该子类的视觉相似的鸟的类别。如果计算出的置信度较低（置信度低于指定好的阈值），我们将会使用一个一比全部的全局SVM分类器。

由于该模型在进行分类预测之前，会先使用聚类的方法将训练或测试的样本分配给相应的子类，从而该模型不用在给数据样本标记要识别的主体进行框选，进一步提高了该方案的可行性。

## 对数据集的选择

本次网络模型的实现采用的数据集是Caltech-UCSD Birds-200-2011 Dataset。Caltech-UCSD Birds-200-2011是用于图像分类的鸟类数据集，它是CUB-200 dataset的一个扩充版本。每个类的图像数量大约增加两倍和新的部位注释。该数据集是细粒度图像分类最广泛使用的基准。该数据集涵盖了200种鸟类，其中包括5994张训练图像和5794张测试图像。除类别标签外，每个图像都会用1个边界框、15个零件关键点和312个属性进行进一步注释。该数据集较为符合本次网络模型实现的要求，因此我们将Caltech-UCSD Birds-200-2011 Dataset作为本次实验的数据集。同时，统计并观察该数据集在搭建的模型上的表现效果。如果有需要，将会考虑人工手动标注一部分数据集，由于本模型不用标注训练或测试样本的各种属性，同时也不用框选训练样本中目标对象的所在的矩形区域，因此数据标注工作相对于其他模型的数据标注工作工作量要小一些。