

北京交通大学

硕士学位论文

企鹅栖息地和数量检测方法的设计与系统实现

Design and System Implementation of Penguin Habitat and Quantity
Detection Method

作者：张宇轩

导师：邢薇薇

北京交通大学

2020 年 6 月

学位论文版权使用授权书

本学位论文作者完全了解北京交通大学有关保留、使用学位论文的规定。特授权北京交通大学可以将学位论文的全部或部分内容编入有关数据库进行检索，提供阅览服务，并采用影印、缩印或扫描等复制手段保存、汇编以供查阅和借阅。同意学校向国家有关部门或机构送交论文的复印件和磁盘。学校可以为存在馆际合作关系的兄弟高校用户提供文献传递服务和交换服务。

（保密的学位论文在解密后适用本授权说明）

学位论文作者签名：张宇轩

导师签名：邵东

签字日期：2020 年 5 月 25 日

签字日期：2020 年 5 月 25 日

学校代码：10004

密级：公开

北京交通大学

硕士专业学位论文

企鹅栖息地和数量检测方法的设计与系统实现

Design and System Implementation of Penguin Habitat and Quantity
Detection Method

作者姓名：张宇轩

学 号：18140691

导师姓名：邢薇薇

职 称：教授

工程硕士专业领域：软件工程

学位级别：硕士

北京交通大学

2020 年 6 月

致谢

首先要感谢我的导师邢薇薇教授，本论文的工作是在邢老师的悉心指导下经过反复推敲完成的。邢薇薇老师在软件行业深耕数年，积累了丰富的经验，在我攻读硕士学位期间，邢老师鼓励我勤于实践、乐于创新，她严谨的治学态度和科学的工作方法给了我极大的影响和帮助。在本人撰写论文期间邢薇薇老师提出了很多宝贵的意见与建议，本人由衷地感激。

在本论文的撰写期间，房子毅、张宇、何冬昕等同学给予本人帮助，对于同学、朋友们的支持与鼓励本人表示感谢。

另外，感谢这些年来所有的对我有过帮助的朋友们和家人们，他们的关爱与无尽支持是我努力前进的动力和方向，没有他们的支持和鼓励，就没有我现在所取得的成就，感谢他们的无私付出。

摘要

在全球变暖的影响下, 南极半岛受到持续“绿化”的生态威胁, 企鹅作为南极地区鸟类生物量最大的物种, 被称为南极环境变化的“生物指示剂”。研究南极典型地区企鹅栖息地分布与种群数量, 为探讨气候变化对企鹅的影响以及对企鹅物种的保护工作提供基础数据和理论指导。因此准确且高效地检测企鹅栖息地和企鹅数量对于企鹅的研究工作具有重大意义。在深度学习快速发展的时代背景下, 利用深度学习技术快速检测不同企鹅岛屿的企鹅栖息地和企鹅数量, 成为准确且高效地获取企鹅相关信息的重要方式。

本文利用目标检测技术对南极典型地区的企鹅栖息地进行检测查找, 确定企鹅栖息地的位置分布; 利用图像语义分割技术对栖息地中企鹅数量进行检测统计, 通过企鹅栖息地与数量检测系统获取企鹅栖息地和企鹅数量的检测结果, 为企鹅和南极生态环境的保护提供数据支持。作者完成了如下工作:

(1) 企鹅岛屿图像的数据预处理。首先通过图像切割、图像批量命名、扩充样本数量和筛选等操作来进行图像数据预处理, 然后对预处理图像进行数据标注后制作成了本文算法模型所需要的数据集。

(2) 基于 Faster RCNN 的企鹅栖息地目标检测方法。针对 Faster RCNN 进行企鹅栖息地检测过程中遇到的漏检和误检问题, 以及对企鹅栖息地进行检测后的展示结果不易辨认等问题, 本文设计并实现一种基于 Faster RCNN 的企鹅栖息地检测方法。在 Faster RCNN 的基础上将 ROI Align 代替 Faster RCNN 中的感兴趣区域池化层来提高 Faster RCNN 对于企鹅栖息地的检测精度, 并通过标注区域图像的方式代替使用检测框标注目标, 使用户更易辨认栖息地检测结果。

(3) 基于全卷积网络 (Fully Convolutional Networks, FCN) 的企鹅数量检测方法。通过将企鹅栖息地检测结果进行处理后作为企鹅数量检测的输入数据, 结合基于 FCN 的企鹅数量检测方法进行检测, 提高了系统从原始图像中检测出企鹅个体并获取企鹅数量的效率。

(4) 企鹅栖息地与数量检测系统的设计与实现。本文将之前提出的企鹅栖息地检测方法与企鹅数量检测方法集成建立了一个企鹅栖息地与数量检测系统。在该系统中导入需要被检测的企鹅岛屿图像, 通过系统处理后可以获得企鹅栖息地与企鹅数量的检测结果, 且可以将检测结果导出查看。

本论文完成的企鹅栖息地与数量检测系统, 通过对系统各模块进行功能测试, 测试结果表明各功能模块都已经达到了预想的效果, 可以为用户高效、准确地提供企鹅栖息地检测数据和企鹅数量数据等信息。

关键词: Faster RCNN; FCN; 目标检测; 语义分割; 企鹅栖息地

ABSTRACT

Under the impact of global warming, the tendency of “greening” has been threatening Antarctic Peninsula. As the largest species in biomass on Antarctic, penguins are known as the bio-indicator of the climate change in Antarctic. Researches in penguins’ population and the distribution of their habitats provide basic data and theoretical guidance to studies in climate change’s impact on penguins and the species’ preservation. Therefore, accurate and efficient monitoring on penguins’ habitats and population is of vital importance to the study of penguins. As deep learning technology develops, monitoring penguins’ habitats and population with this new technology has become an important approach of acquiring relevant information.

The researcher determined the geographic distribution of penguins’ habitats through object detection technology and conducted test statistics on penguins’ population in habitats through semantic segmentation, in turn, results of penguins’ habitats and its population was obtained by quantity surveying, offering statistic backup for preservation of penguins and Antarctic’s environment. Steps to fulfill the task are as follows.

(1)Preprocessing of data drew from graphics of penguins’ island habitats. Preprocessing involves image segmentation, bulk image naming, expanding and sorting sample quantity and so forth. Preprocessed images, after data annotation, became data set needed for algorithmic model of this research.

(2)The object detection of this research was based on Faster RCNN. Considering the missed and false detection during the process and the difficulty in recognizing the results, the researcher designed and realized a detecting method on the basis of Faster RCNN, which replaces the region of interest pooling in Faster RCNN with ROI Align, increasing the accuracy of detecting. Besides, this method makes users recognize results more easily by replacing detection box labelling with region image labelling.

(3)The detecting method based on Fully Convolutional Networks (FCN). Conducting the detection by inputting the processed detecting results of penguins’ habitats, combining with FCN technology, the efficiency of extracting individual penguins from original images was improved.

(4)The design and realization of penguins’ habitats and population detecting system. Based on methods mentioned above, the researcher built a system to detect penguins’

habitats and population. The system can produce detecting results of penguin habitats and population by inputting images needed for detecting.

After testing modules of the system, the results showed all modules have reached desirable effectiveness. Therefore, it can provide data on penguin habitats and population for users with high efficiency and accuracy.

KEYWORDS: Faster RCNN; FCN; Object Detection; Semantic Segmentation; Penguin habitat

目 录

摘要.....	III
ABSTRACT.....	IV
1 引言.....	1
1.1 论文背景与意义.....	1
1.2 国内外发展现状.....	1
1.3 论文的主要内容.....	3
1.4 论文组织结构.....	4
1.5 本章小结.....	5
2 关键理论及技术综述.....	6
2.1 目标检测算法.....	6
2.1.1 RCNN.....	7
2.1.2 SPPNET.....	7
2.1.3 FAST RCNN.....	7
2.1.4 FASTER RCNN.....	8
2.2 CNN 与 FCN 基本原理.....	10
2.2.1 CNN.....	10
2.2.2 FCN.....	12
2.3 本章小结.....	14
3 企鹅岛屿图像的数据预处理.....	15
3.1 相关数据介绍.....	15
3.2 数据预处理.....	16
3.2.1 图像切割.....	16
3.2.2 图像批量命名.....	17
3.2.3 扩充样本数量.....	17
3.2.4 数据筛选.....	17
3.3 数据标注.....	18
3.4 数据转换.....	20
3.5 本章小结.....	20
4 基于 FASTER RCNN 的企鹅栖息地检测方法.....	21

4.1	设计目标.....	21
4.2	改进的 FASTER RCNN.....	22
4.2.1	ROI ALIGN.....	22
4.2.2	优化展示结果.....	24
4.3	实验设计与结果分析.....	25
4.3.1	实验数据集和实验配置.....	25
4.3.2	评价指标.....	26
4.3.3	实验结果与分析.....	27
4.4	本章小结.....	28
5	基于 FCN 的企鹅数量检测方法.....	30
5.1	设计目标.....	30
5.2	数据与方法.....	31
5.2.1	数据集描述.....	31
5.2.2	企鹅检测与数量统计方法.....	31
5.3	实验设计与结果分析.....	33
5.3.1	模型训练过程.....	33
5.3.2	评价指标.....	33
5.3.3	实验结果与分析.....	33
5.4	本章小结.....	35
6	企鹅栖息地与数量检测系统的设计与实现.....	36
6.1	系统需求分析.....	36
6.1.1	需求分析综述.....	36
6.1.2	功能性需求.....	37
6.1.3	非功能性需求.....	38
6.2	系统概要设计.....	39
6.3	系统功能模块设计与实现.....	41
6.3.1	图像上传模块.....	42
6.3.2	企鹅栖息地检测模块.....	44
6.3.3	企鹅数量检测模块.....	45
6.3.4	检测对比模块.....	47
6.4	系统测试.....	49
6.5	本章小结.....	50

7 结论.....	51
7.1 论文总结.....	51
7.2 工作展望.....	52
参考文献.....	53
作者简历及攻读硕士学位期间取得的研究成果.....	55
独创性声明.....	56
学位论文数据集.....	57

1 引言

本章作为全文的引言，首先对企鹅栖息地与数量检测系统的背景及意义进行介绍，讨论在南极生态环境变化愈发迅速的情况下，准确并高效地检测出企鹅栖息地和企鹅数量对保护企鹅和南极生态环境具有重要意义。然后作者对国内外关于企鹅栖息地与数量检测的相关发展现状进行阐述，并对本论文主要工作内容进行介绍，在本章的最后部分对本论文的整体组织结构进行梳理。

1.1 论文背景与意义

南半球海冰面积每年都以极快的速度消减，达到了 40 年来南半球海冰覆盖面积的最低点，相比同期北极快速融化地区，南极地区的海冰面积降幅要超过了北极地区的降幅^[1]，种种迹象都在说明南极冰盖正在加速融化，情况不容乐观。在全球变暖的影响下，未来南极生态系统将发生剧烈的变化，南极地区的典型生物和标志性地形也会受到巨大影响。依赖海冰生存在南极半岛的帝企鹅和阿德利企鹅在南极海冰逐渐消退的影响下，其繁殖栖息地越来越少，成为南极气候变化影响下的受害者。自 2018 年初以来，南极生态研究人员在调查研究中发现南极洲大部分冰雪样品中都存在微塑料和有害化学物质。全球变暖导致海冰面积逐年减少，再加上环境污染的影响，使本就脆弱的南极洲生态系统面临更大的威胁。

在地球上现存的 18 种企鹅中，有 11 种企鹅种群的数量正在不断减少，其中有 10 种企鹅种群被归类为“脆弱”或“濒危”的情况^[2]，这些现状表明企鹅的整体数量正在不断地减少并且需要人类对其的关注和保护。同时企鹅作为南极地区最大鸟类生物量的物种^[3]，一直被视为南极陆地和海洋栖息地生态健康的主要生物指示鸟^[4]。随着南极生态环境变化速度越来越快，对南极典型地区在不同气候变化下的企鹅栖息地分布与企鹅数量研究工作，可以为企鹅物种的研究与保护提供数据支持和理论指导。因此准确且高效地完成企鹅栖息地和企鹅数量检测对于企鹅的研究工作具有重大意义。

1.2 国内外发展现状

遥感作为一项用于监测物种丰富度和物种分布区域的重要技术，在对全球范围内的气候变化和环境变化的研究中也经常被使用。在过去对企鹅栖息地的位置

分布和企鹅数量的获取方式,主要是通过卫星图像作为探测和监测手段^[5],来获取南极岛屿的图像信息。对于南极企鹅栖息地的部分区域,因为技术限制和研究资源限制等条件的影响使企鹅研究人员无法进入,这就需要依靠卫星图像作为探测和监测企鹅数量的重要数据采集途径。早在上世纪 80 年代,陆地卫星首次展示了企鹅栖息地的遥感影像,企鹅栖息地的粪便可以与周围的景观区分开来,通过观测企鹅粪便区域来判断企鹅栖息地。随着遥感数据的发展,多种航空照片被用来确定企鹅数量及其栖息地分布的变化,通常使用中分辨率卫星遥感数据对企鹅栖息地进行检测^[6],使用高分辨率卫星数据识别企鹅种群及种群对应的数量^[7]。

以前基于卫星拍摄的企鹅影像资料,对企鹅进行研究的過程中,多数情况下是通过目视解译或监督分类等方法对影像中的企鹅粪便区域进行解译提取处理,在这基础上通过建立企鹅粪便区面积与企鹅数量回归关系的方式来获取企鹅数量^[8]。但是在对企鹅粪便区面积进行提取的过程中,因为提取误差以及企鹅密度分布不均等问题的存在,会对该方式获取企鹅数量信息的准确性有较大影响。而且在过去做数量统计时,使用目视解译等方式对人力消耗的成本较高,且统计过程中效率低下,手动标注企鹅群体通常情况下需要消耗大量的时间和精力。对于如此费力的统计方式,再加上研究过程中需要对大量的图像进行标注来获取足够的数据集,以及南极生态环境变化和企鹅物种变动越来越快,需要更加准确且高效的方法获取企鹅相关研究信息。

之前多采用监督分类和非监督分类作为基于像元的分类方法,来对遥感影像进行分类。但由于空间分辨率的限制,该方法过分依赖于图像的光谱信息,导致其分类结果中存在较多分类错误和遗漏的问题^[9]。此时面向对象的图像分类技术被提出,用来解决传统遥感图像分类技术中存在的问题^[10],该技术通过借助高分辨率遥感影像中,相较于中分辨率遥感影像更为丰富的空间信息来实现更好的分类效果。

Witharana 等人^[4]在对金图企鹅与阿德利企鹅粪便进行区分的实验结果中,通过使用面向对象法处理高分辨率遥感影像,达到了较好的分类效果。贺红等人^[11]采用面向对象法对南极维多利亚地罗斯海地区的难言岛航拍图像进行识别,得到了岛上阿德利企鹅的数量信息,该实验结果表明面向对象法使用彩色影像在识别精度的效果上要优于黑白影像^[12]。由于中高分辨率遥感影像的分类工作仍然需要借助人工目视解译,且在企鹅登陆繁殖的季节,遥感影像的获取与解译容易受到云层的影响。近年在高分辨率卫星影像和航空摄影遥感技术加速发展的时代背景下,对南极企鹅研究的制图精度得到明显提高,利用企鹅像元直接研究企鹅种群分布成为可能。无人机由于其体积小、操作方便和分辨率高等优点,近年来常用于南极企鹅种群监测,打破了传统遥感方法的限制^[13]。

Goebel 等人^[14]使用无人机对南极地区的企鹅数量进行统计, 并通过大量的实验来验证该方法的可行性, 冀明等人^[13]基于高精度无人机航拍影像与面向对象法的结合, 对阿德雷岛等南极区域岛屿进行企鹅栖息地与数量检测, 取得的识别效果证明无人机航拍影像在南极生态应用中具有巨大价值。

随着互联网与人工智能的发展, 近年计算机的计算能力也在飞快提升, 深度学习被广泛应用于各工程领域。目标检测技术作为如今计算机视觉处理的一个重要应用对象, 被广泛应用于人脸识别领域、导航领域、图像和视频的监控与检测等多方面领域^[15], 该技术实现了通过计算机视觉, 来减少人工参与图像检测, 对降低人力成本的消耗具有重要意义。

卷积神经网络作为计算机视觉领域的另一个重要应用对象, 在进行部分领域内的图像处理时, 其分类结果准确性强于人类的识别能力。卷积神经网络因为其强大的性能, 在图像识别领域具有非常重要的研究意义和价值, 目前被广泛应用于文字识别分割、人脸识别和车辆识别等领域^[16]。在卷积神经网络的基础上, 全卷积神经网络做出部分改进, 实现了端到端、像素到像素的图像处理, 而且相比卷积神经网络, 全卷积神经网络不存在对输入图像尺寸的限制, 全卷积神经网络可以实现输入任意大小的图像, 在图像处理的灵活性和效率上都有很大的提高。目前国内外已经将全卷积神经网络广泛应用于图像处理领域, 如医学图像和卫星航拍识别等^[17]。

1.3 论文的主要内容

本论文通过结合目标检测技术和图像分割技术, 来统计南极区域各个岛屿企鹅栖息地和数量分布, 以达到降低人力成本的消耗并提高统计效率的目的。本人主要完成以下工作。

(1) 企鹅岛屿图像的数据处理工作。首先对本论文所需数据集和图像预处理方法进行设计。通过图像切割、图像批量命名、扩充样本数量和筛选等操作来进行图像数据预处理, 并通过数据标注预处理图像制作成了本文算法模型需要的数据集。因为本文使用了两种算法来分别实现对应的检测功能, 所以创建的数据集为两组, 分别为本文两种算法模型提供数据集基础。

(2) 基于 Faster RCNN 的企鹅栖息地目标检测方法的设计。针对 Faster RCNN 在实现企鹅栖息地检测的过程中遇到的漏检、误检问题, 以及对企鹅栖息地检测后的展示结果不易辨认等问题, 对企鹅栖息地检测方法进行改进。在 Faster RCNN 基础上将 ROI Align 代替 Faster RCNN 中的感兴趣区域池化层来提高 Faster RCNN 对企鹅栖息地检测的精度, 并通过取消检测框标注目标, 采用对检测目标所在切

割小图进行整体标注的方法,更好地对检测结果进行展示,而且做出这样的设计后省去了计算边框回归偏移量这一过程,间接提高了算法整体检测速度。

(3)基于 FCN 的企鹅数量检测方法的设计。通过将之前企鹅栖息地检测方法获取的检测结果,经过处理后作为企鹅数量检测的输入数据,结合基于 FCN 的企鹅数量检测方法进行检测,提高了系统从原始图像中检测出企鹅个体的效率。

(4)企鹅栖息地与数量检测系统的设计与实现。本文将之前提出的企鹅栖息地检测方法与企鹅数量检测方法集成起来,建立了一个企鹅栖息地与数量检测系统。通过该系统,用户可以导入需要被检测的企鹅岛屿图像,通过系统处理后可以获得企鹅栖息地与企鹅数量的检测结果,且可以将检测结果导出查看。

(5)完成该系统的功能测试并对出现的系统问题进行修复。

1.4 论文组织结构

根据实际研发情况,本论文共分为七个章节,从不同的角度对论文进行描述。章节的具体内容如下:

第一章绪论。本章节分为四部分来对本论文整体情况进行介绍,首先对本论文的背景与意义进行阐述,在此基础上对企鹅栖息地检测与企鹅数量检测技术发展现状进行描述与分析,最后对本论文的主要内容和论文框架进行了详细介绍。

第二章关键理论和技术综述。该章节主要对本论文使用算法的相关理论知识进行梳理和介绍,首先介绍了用于企鹅栖息地检测的 Faster RCNN 目标检测算法的发展历程,并对 Faster RCNN 目标检测算法的技术特点进行阐述。然后介绍了卷积神经网络和用于企鹅数量检测的全卷积神经网络,并对后者技术特点进行介绍。

第三章图像的数据预处理。主要对数据集和图像预处理方法进行设计。首先,通过图像切割、图像批量命名、扩充样本数量和筛选等操作来进行图像数据预处理,并通过数据标注预处理图像制作成了本文算法模型需要的数据集。

第四章基于 Faster RCNN 的企鹅栖息地检测方法。针对 Faster RCNN 在实现企鹅栖息地检测的过程中遇到的漏检、误检问题,以及对企鹅栖息地检测后的展示结果不易辨认等问题,对企鹅栖息地检测方法进行了改进。

第五章基于 FCN 的企鹅数量检测方法。通过将基于 Faster RCNN 的企鹅栖息地检测方法获取的企鹅栖息地检测结果,经过处理后作为企鹅数量检测的输入数据,结合基于 FCN 的企鹅数量检测方法进行检测。

第六章企鹅栖息地与数量检测系统的设计与实现。基于之前的设计和工作,将以上提出的企鹅栖息地检测方法与企鹅数量检测方法集成起来,建立了一个企

鹅栖息地与数量检测系统。本章对企鹅栖息地与数量检测系统的整体设计方案和系统各组成模块的设计思路进行介绍，并分别对系统各模块的功能与实现进行阐述。最后使用阿德雷岛图像对系统进行性能测试，得到并展示测试结果。

第七章结论。该章节对本论文的全部工作进行了总结，整理企鹅栖息地与数量检测方法的设计与系统实现的工作内容，对本论文中可以进行改进的点和不足之处进行分析和总结，最后提出对未来工作的展望。

1.5 本章小结

本章首先介绍了南极环境现状以及企鹅生存现状等背景，以及进行企鹅栖息地与企鹅数量检测研究的价值和意义，其次分析了现在企鹅栖息地与企鹅数量检测技术的国内外发展现状，最后对本论文的主要内容和论文组织结构进行了详细介绍。

2 关键理论及技术综述

本章主要介绍企鹅栖息地检测和企鹅数量检测使用的算法。首先介绍用于企鹅栖息地检测的 Faster RCNN 目标检测算法的发展历程，并对 Faster RCNN 目标检测算法的技术特点进行阐述。然后介绍了卷积神经网络和用于企鹅数量检测的全卷积神经网络，并对后者的技术特点重点介绍。

2.1 目标检测算法

目标检测 (Object Detection) 是在计算机视觉相关领域中具有较高实用性和挑战性的任务。目标检测的任务有两大类，分别为目标实例检测 (Instance Object Detection) 和目标类别检测 (Generic Object Detection)。其检测任务目标是解决“在哪里”和“是什么”整个问题，简单理解为图片分类与目标定位的综合，即在一张图片中识别出目标和其所在的精确位置，并标识出目标类别。传统的目标检测方法根据其检测过程的特点，一般被分为三个阶段。

第一阶段采用滑动窗口遍历整幅图像来进行区域选择。因为检测的目标可能会在图像上任意位置出现，所以此阶段在给定的图像上选择一些候选区域，目的是定位图像上目标出现的位置。

第二阶段利用 SIFT 和 HOG 等特征进行图像特征提取。因为目标形态的不确定性，需要对选择的区域进行特征提取的操作。特征提取的效果好坏会直接对目标分类的准确性造成影响。

第三阶段采用预先训练好的 SVM、Adaboost 等分类器进行分类。

因为使用滑动窗口来进行区域选择的策略针对性很差，并且遍历过程的时间复杂度和窗口重复率都很高，再加上通过手工设计的特征很难应对多样、复杂的影响因素，传统目标检测技术具有极大优化空间。

目前基于深度学习的目标检测方法根据检测过程的特点可以分为两大类。一类是基于回归方法的 One-Stage 类目标检测算法^[18]；另一类是基于候选区域的 Two-Stage 类目标检测算法^[19]。由于在企鹅栖息地检测过程中使用的算法为 Faster RCNN，这里对基于候选区域的 Two-Stage 类目标检测算法的发展阶段进行简单介绍，然后在此基础上对 Faster RCNN 进行详细的介绍。

2.1.1 RCNN

RCNN (Regions with CNN features) [20]是最早将卷积神经网络 (Convolutional Neural Networks, CNN) 作为处理目标检测相关问题上的算法里程碑, 其方法是基于传统目标检测方法来找出一些可能是目标所在位置的候选区域, 通过将大概率为目标区域的尺寸大小进行归一化成为卷积网络输入的尺寸大小, 进而来进行判断该区域是不是目标, 是哪类目标。对于确定是目标的区域进行再次回归的调整学习, 使得选中目标的概率更高。在此过程中 RCNN 通过使用 CNN 进行目标特征提取, 以此来取代传统的特征提取方式, 减少了特征提取过程中需要消耗的时间, 最后通过 SVM 和 Softmax 分类器实现对目标的分类识别过程。RCNN 的缺陷是对输入图像的尺寸有严格要求, 需要固定大小的尺寸, 这就有可能影响尺寸不确定图像的识别精度。

2.1.2 SPPNet

Kaiming He 等人 [21]在 RCNN 的基础上做出了实质性的改进, 研究出更具原则的网络池化策略, 叫做 SPPNet (Spatial Pyramid Pooling Network)。SPPNet 主要解决了深度网络固定输入层尺寸的限制问题, 取消了图像归一化的过程, 也从多方面表明了不限制输入尺寸所带来的优势, 并且 SPPNet 使用空间金字塔池化网路的结构代替了全连接层前的最后一个池化层, 使得处理一张图片仅需要对其卷积层特征进行一次提取过程, 最后在全连接层中输入每个候选区域的卷积层特征来完成后续步骤。相较于 RCNN, SPPNet 虽然对目标检测速度的提高具有较大促进作用, 但是仍存在算法步骤复杂的情况, 且只是对全连接层进行很少的调整。

2.1.3 Fast RCNN

Ross B. Girshick [22] 在 2015 年提出了 Fast RCNN (Fast Region-Based Convolutional Network) 算法。Fast RCNN 在 RCNN 的基础上进一步提升了目标检测的整体计算速度, 使用 CNN 对目标候选区域进行分类。与之前相比, Fast RCNN 在训练过程和测试过程中有许多优化点, 对检测的准确率有较大提升。

其特点之一为在最后一个卷积层后增加了感兴趣区域池化层 (Region of interest pooling layer, ROI Pooling layer) 的同时, 也加入了使网络能够反向传播的候选框映射功能; 另一个特点是使用了多任务损失函数, 在其卷积神经网络中加入边框回归进行训练。Fast RCNN 很大程度上对 SPPNet 与 RCNN 的优点进行结合,

加入多任务损失函数，对算法流程进行简单化处理，使训练过程和测试过程变得更加便利，有实验表明 Fast RCNN 在训练速度上相比 RCNN 快 9 倍多，在测试速度上要快 213 倍多^[23]。但由于其使用选择性搜索（Selective Search），导致消耗大量时间提取候选区域，对于特征分类的提取却占用很少时间，因此在实际应用过程中对目标检测的实时性要求无法得到保障，Fast RCNN 也没有从真正意义上实现端到端的训练过程。

2.1.4 Faster RCNN

Faster RCNN（Faster Region-Based Convolutional Network）是由 Ross B. Girshick 在 2016 年提出的一种基于候选区域的目标检测方法，该方法也是目前具有较高准确率的在目标检测模型之一。从 RCNN 到 SPPNet，再发展到 Fast RCNN，经过在先前基础上一系列的进步，发展到了如今的 Faster RCNN，候选区域生成、特征提取、分类、位置精修这四个基本步骤终于在 Faster RCNN 上进行了统一，并且保证了这些步骤不发生重复计算，对整体运算速度的提高效果非常显著。

相较于 Fast RCNN，Faster RCNN 可以被简单理解为区域生成网络（Region Proposal Network, RPN）和 Fast RCNN 的结合体，因为在 Faster RCNN 中采用区域生成网络对 Fast RCNN 的选择性搜索方法进行替代，这也是二者的主要区别。有实验表明这一改进对单张图片的候选区域提取速度达到 10 毫秒，比改进之前的速度快了 10 倍，可以确定该方法对目标检测的速度进行了大幅提升。如图 2-1 所示。图片中分别对 Fast RCNN 和 Faster RCNN 的候选区域生成过程进行展示，Faster RCNN 在对候选区域进行提取时使用区域生成网络代替选择性搜索方法。

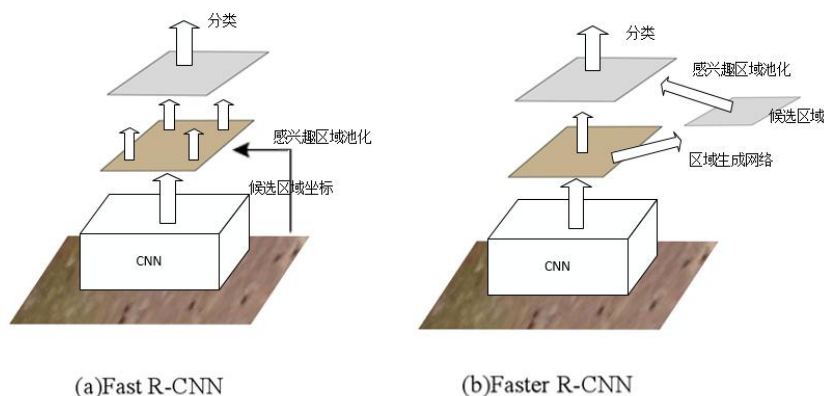


图 2-1 Fast RCNN 与 Faster RCNN 的候选区域生成过程

Figure 2-1 The process of region proposal generation by Fast RCNN and Faster RCNN

如图 2-2 所示, 该算法的处理流程非常清晰, 每一步骤都不会发生重复计算。首先将要进行检测的图像输入到 CNN 中进行图像的提取特征, 接着通过 RPN 网络来生成目标图像的候选区域, 通过感兴趣区域池化层对每个候选区域都生成固定尺寸大小的特征图后, 使用 Softmax 分类器对目标类别和目标具体的边框区域进行计算, 得到目标识别和位置回归。

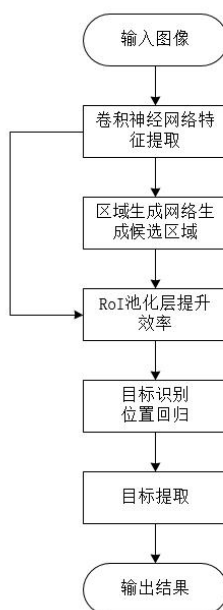


图 2-2 Faster RCNN 的处理流程
Figure 2-2 Faster RCNN processing flowchart

检测企鹅栖息地对目标检测算法的精度要求很高, 而 Faster RCNN 的检测平均精度是非常高的, 综合考虑当前热门的目标检测算法, 最终选择使用 Faster RCNN 对企鹅栖息地进行检测。作为 Faster RCNN 的核心, 区域生成网络替换选择性搜索使算法更加快速, 我们对其工作流程进行简单介绍。

首先 Faster RCNN 通过 13 个 conv 层、13 个 relu 层和 4 个 pooling 层来进行图像的特征提取, 对于 conv 层和 relu 层来说, 并不会改变图像在提取特征过程中的输入和输出大小, 只有 pooling 层使输出图像的长宽都变为输入的 1/2, 由于 4 个 pooling 层的处理影响, 图像矩阵经过特征提取后为原图矩阵的 1/256, 这样生成的特征映射也可以和原图对应起来。

区域生成网络的输入就是图像的特征映射, 每一个特征映射在 RPN 中都会配备 9 个锚点 (anchor) 作为其初始检测框, 每一个锚点的四个值表示锚点矩形框的左上角和右下角的坐标。而矩形框有三种形状, 分别对应长宽比为 1:1、1:2、2:1 的矩形, 这样基本上就能覆盖图像的各个尺度和形状。此时的由锚点表示的初始检测框虽然不够精准, 但是可在后续过程中进行修正。可以简单理解为 RPN 的作

用就是在原图尺寸上设置了密密麻麻的锚点,然后利用 softmax 分类器去判断锚点内是否有目标,将包含目标的锚点标识为正样本,不包含目标的锚点标识为负样本,然后将提取出正样本锚点作为候选区域。

提取候选区域的工作完成后,接下来进行修正候选框,通过边界回归(Bounding box regression)对边框的平移和缩放过程使得输入原始正样本锚点经过映射得到跟真实目标更接近的回归窗口,然后将修正过程计算得到的变换量和正样本锚点进行综合处理后,生成更加精确的候选框,送入到后续的感兴趣区域池化层中。

将 Faster RCNN 模型实际应用到企鹅栖息地检测中,发现了两个问题:(1)经过初次实验发现有一些小目标的栖息地并没有被检测出来,因此重新对特征点进行标注,增加小区域目标后而发现了大量的误检,在海洋和山丘等地区出现了非常多的误检区域。(2)在检测结果展示中,由于企鹅聚集分布不均匀,而且相对于整个岛屿图像来说,对切割小图进行检测得到的检测框太过微小,由此明确对于企鹅栖息地的检测结果并不是要求使用检测框对图像中的目标进行标记,而是需要更明显的标记方式来确认企鹅栖息地的分布。

基于上述原因,本文对 Faster RCNN 进行改进调整,首先将 ROI Align (Region of interest align)^[24]代替 Faster RCNN 中的感兴趣区域池化层来提高 Faster RCNN 对于企鹅栖息地的检测精度,解决了实际场景中检测结果不明确、漏检小目标栖息地以及误检区域过多等问题的困扰,进一步提高了企鹅栖息地检测的准确率。然后通过取消检测框标注目标,采用对检测目标所在切割小图进行整体标注的方法,不但可以更明显的在岛屿图像中分辨出企鹅栖息地,而且检测出包含企鹅的切割小图可直接作为企鹅数量检测的输入数据。做出这样的设计后省去了计算边界回归偏移量这一过程,间接提高了算法整体检测速度。

2.2 CNN 与 FCN 基本原理

本节主要对 CNN 与 FCN 的基本原理进行介绍,并通过对二者部分细节的阐述,明确了选取 FCN 作为企鹅数量检测模型的理由,最后对 FCN 的实现流程做出阐述。

2.2.1 CNN

卷积神经网络^[25]最早是由 Yann Lecun 于 90 年代提出的前馈神经网络。经历了 10 年的沉寂后,2012 年 AlexNet 以较高的成绩优势夺冠后开启了深度学习的热潮。在一些图像识别的测试中,卷积神经网络的表现出来的能力极其强大,而且在语音识别和自然语言处理等领域也有着不俗的表现。CNN 的基本框架主要为三部分:

卷积层（Convolutional layer）、池化层（Pooling layer）和全连接层（Full-connected layer），以下对 CNN 的基本框架进行详细介绍。

(1) 卷积层

卷积层作为卷积神经网络的核心，在图像识别领域其作用为识别图片中的指定特征，其操作为通过卷积核在原图上采用滑动窗口进行卷积运算，完成特征提取，得到特征映射（Feature map）。卷积核是在卷积过程中通过训练学习得到的参数，而卷积层的参数是由卷积层的两个主要特点决定的，即局部连接和权值共享。局部连接的思想源自于生物学中视觉皮层神经元，特征图上的每个值只对应着原图一小部分区域，可以看做每个神经元只是与输入神经元的一块区域连接，连接方式就是局部连接，而原图上的这块局部区域被称为感受野（Receptive field）。对于图像识别来说局部像素之间也会有一定的关联性，每个神经元只需要感受局部特征而非对全部图像做感受，就可以通过更高层将不同局部神经元综合起来获取全局信息。权值共享是指因为不同神经元可以共享权值，可以达到减少求解参数的目的。在卷积核对整张图片进行处理的过程中，卷积核参数可以不发生改变，它在一个区域中提取到的特征也能适用其他区域，是因为其可以检测到不同位置的同一类型特征，也就说明卷积网络能很好的适应图像的小范围的平移性，即有较好的平移不变性。

(2) 池化层

池化层的主要功能是在保留图片特征的同时可以压缩数据量，池化层处在卷积层之后进行操作，常见的池化方式分为最大池化（Max-pooling）和平均池化（Avg-pooling），最大池化的输出是通过提取不同区域的最大值得到的，平均池化的输出是通过提取每个区域的平均值得到的。池化的本质是下采样，特征图经卷积层的输出后作为输入特征进行特征选择和特征降维，在保留特征图特征的同时实现了数据压缩。

(3) 全连接层

全连接层在卷积神经网络中的作用是为了进行分类，在全连接层之前的卷积层和池化层使原始图像数据映射到特征空间，而全连接层在卷积神经网络的最后几层中对前面得到的特征进行加权以减少特征信息的丢失，实现将特征空间映射到样本空间。

CNN 的优势是其多层结构可以实现自动学习这一特征，而且能够学习到多个层次特征。其中对于较浅的卷积层由于其感知域较小，CNN 可以学习到其中的局部区域特征；对于较深的卷积层拥有较大的感知域，CNN 能够通过学习得到更加抽象的特征。抽象特征由于其对目标图像的大小、位置和方向等敏感性较低的特性，更利于提高识别性能。

在使用 CNN 对一个像素进行分类的过程中,通过将该像素周围的一个图像块作为 CNN 的输入用于训练和预测来体现 CNN 的几个缺点。首先是对存储空间的需求很大。假如使用的大小为 15×15 的图像块作为输入,在进行滑动窗口的过程中,因为每次滑动的窗口都会给 CNN 进行分类判别,导致所需要的存储空间会根据滑动窗口次数和大小的变动而急剧上升。其次是计算效率低。因为相邻的像素块重复率很高,如果对每个像素块都进行卷积的计算,那么计算过程大概率也是重复的。第三点是像素块的大小会对感知区域的大小有一定限制。正常情况下像素块大小会比整幅图像小很多,因此对像素进行特征提取的话,只能提取有限的局部特征,在一定程度上限制分类的性能。全卷积神经网络可以从抽象的特征中对每个像素的类别进行恢复,实现了像素级别的分类而非图像级别的分类。

2.2.2 FCN

传统的图像分割方法是先对图像中的一个像素点进行选取,然后通过在该像素周围划分出对应图像块,并将该图像块的特征作为样本对分类器进行训练,判断像素点的结果就是分类得到的结果。由此可见传统方法存在很多缺陷,比如无法对图像块的大小进行确认,如果图像块太小的话,有可能很难获取其目标特征信息,并且需要消耗的时间量也很大。

CNN 网络通过在其卷积层之后接上全连接层的方式将卷积层产生的特征映射处理成固定长度的特征向量,来得到整个输入图像的数据描述。该数值描述代表着输入图像的类别概率,是对图像级别的分类。图 2-3 展示了卷积神经网络如何实现分类。

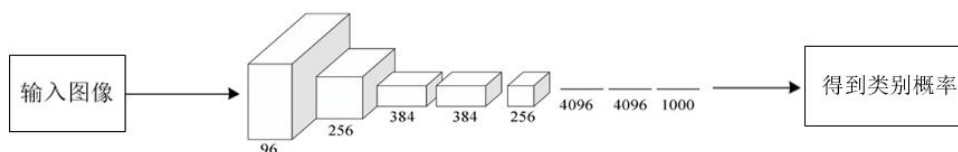


图 2-3 卷积网络实现分类过程

Figure 2-3 The implementation classification process of CNN

与全连接层相比,卷积层的特点在于与它的神经元进行连接的只是输入数据中的一个局部区域,同时神经元可以在卷积层中实现对参数的共享。由于全连接层和卷积层的神经元共性是二者都对点积进行计算,这样保证了二者函数形式的差异不大,从而可以实现对全连接层和卷积层的相互转化。

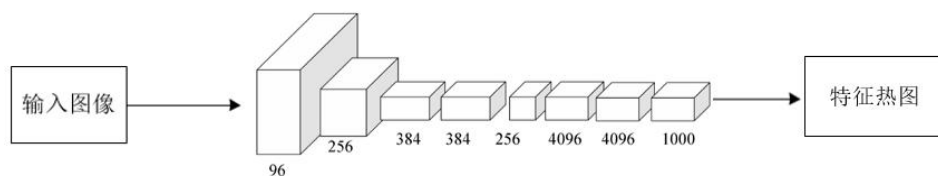


图 2-4 全卷积网络实现分类过程
Figure 2-4 The implementation classification process of FCN

如图 2-4 所示。FCN 在 CNN 的基础上进行改进，将 CNN 网络中的后三层全连接层转化成三层卷积层，改进后可以做到不限制输入图像的大小。

输入原图经过在 FCN 中进行多次卷积和池化之后，使图像和图像对应的分辨率都逐渐减小，当处理到最后一层时，图像尺寸变为 $H/32 \times W/32$ （ H 和 W 分别代表原图尺寸的高度和宽度），此时产生的高维特征图叫做热图（Heatmap），在此过程中实现了对每个像素都进行了分类预测，并将预测结果标记在每个对应像素点上。完成以上内容后开始对热图进行上采样（Upsample）操作，实现对特征图像的放大过程，最终将特征图像放大到与原图像的尺寸相同，以此达到 FCN 的输出图像要求。

在对输入图像进行第一次卷积和池化后，得到的图像缩小为原图像大小的 $1/2$ ；在此基础上对得到的图像进行第二次卷积和池化，此时得到的图像缩小为之前得到图像的 $1/2$ ，也就是原图像尺寸的 $1/4$ ；同样的操作对上一次得到的图像进行第三次、第四次和第五次的卷积与池化处理，分别得到的图像尺寸为原图尺寸的 $1/8$ ， $1/16$ 和 $1/32$ 。通过第六层和第七层卷积层对得到的缩小为原图 $1/32$ 尺寸大小的图像进行卷积处理，得到的高维特征图像就是上文提到的热图。

通过多层卷积层的处理得到的有原图 $1/32$ 尺寸大小的热图和分别为原图 $1/16$ 、 $1/8$ 、 $1/4$ 、 $1/2$ 尺寸大小的多个特征图。在对热图进行上采样操作后，还原的图像特征只包含第五层卷积层处理的特征，得到 FCN-32s。由于受精度的影响，此时得到的图像无法准确还原输入图像的特征。因此 FCN 中设计了跳跃结构（Skip Architecture），将在此时对上采样处理的结果进行前向迭代，把第四层卷积层处理的特征与上一次上采样操作后得到的特征图像进行反卷积细节补充，得到 FCN-16s；把第三层卷积层处理的特征与上一次上采样操作后得到的特征图像进行反卷积细节补充，得到 FCN-8s，还原的图像精准度也越来越高。FCN 可以处理任意大小的图像，将特征图调整到原图尺寸后计算每个像素对应的类别，放大后的特征图可以保持原图中的位置信息。图 2-5 展示了使用不同特征层上采样融合的结果。

通过不同上采样结构得到的结果对比如图 2-6 所示^[26]。FCN 通过为每个像素分配标签实现了语义层级的分割效果。它的优势是通过使用反卷积、上池化等采

样操作将特征图恢复到原图像大小，然后判断各个位置的像素所属的类别进而确定物体的边界位置。使用 FCN 对目标进行检测时不使用候选区域计算物体的边框，而是在密集的特征图上直接确定物体的边界。由于不需要设计不同宽高比的锚盒，定位形状不规则的物体时可以更好的贴合目标轮廓。FCN 用于实现目标分割的特征图的分辨率最高，而在企鹅数量检测中需要借助企鹅轮廓来区别企鹅和背景，因此最终选择使用 FCN 获取企鹅数量信息。

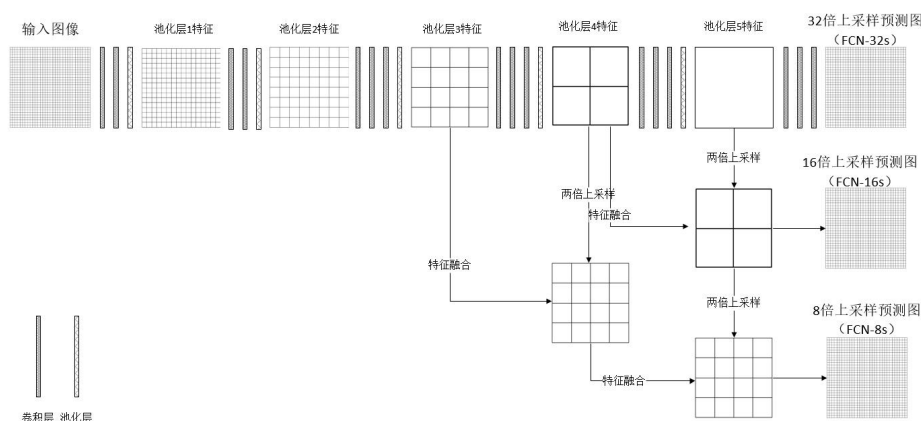


图 2-5 FCN 网络跳跃结构
Figure 2-5 FCN Skip Architecture

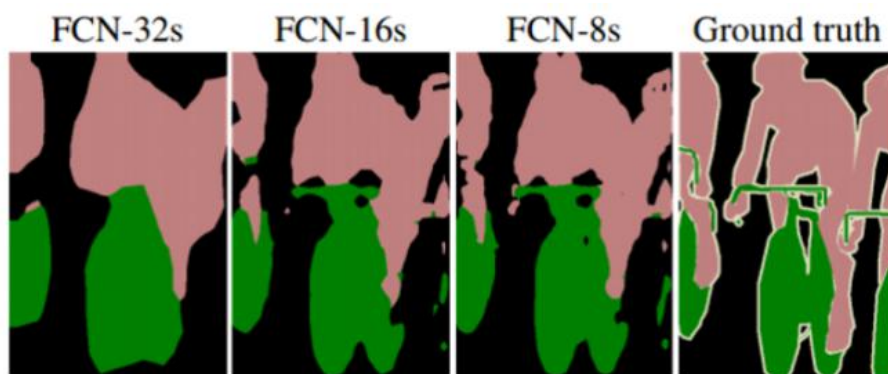


图 2-6 不同上采样结果对比^[26]
Figure 2-6 Comparison of different upsampling results^[26]

2.3 本章小结

本章主要介绍本论文用于企鹅栖息地检测和企鹅数量检测的算法和算法相关内容的介绍，首先介绍了用于企鹅栖息地检测的 Faster RCNN 目标检测算法的发展由来，并详细介绍 Faster RCNN 的技术特点。然后介绍了卷积神经网络和用于企鹅数量检测的全卷积神经网络，并详细介绍后者的技术特点。

3 企鹅岛屿图像的数据预处理

本章主要对本论文数据集的图像数据预处理方法进行设计。通过图像切割、图像批量命名、扩充样本数量和筛选等操作来进行图像数据预处理，并通过对预处理后的图像进行数据标注制作成了本文算法模型需要的数据集。

3.1 相关数据介绍

本文以南极地区的阿德雷岛、拜尔斯半岛以及中山站附近的企鹅岛这三座岛屿的遥感影像数据作为企鹅栖息地检测算法与企鹅数量检测算法的数据集，以上数据集使用的三幅遥感影像数据均来自北京师范大学全球变化与地球系统科学研究院极地遥感团队在中国第 32 次南极科学考察期间，以及和中智国际合作科考期间使用固定翼无人机航拍得到的影像。航拍影像通过镶嵌拼接与正射校正之后，得到企鹅栖息地的影像资料图。



图 3-1 南极阿德雷岛影像资料
Figure 3-1 Image Data of Adelaide Island in Antarctica



图 3-2 拜尔斯半岛影像资料
Figure 3-2 Image Data of Byers Peninsula

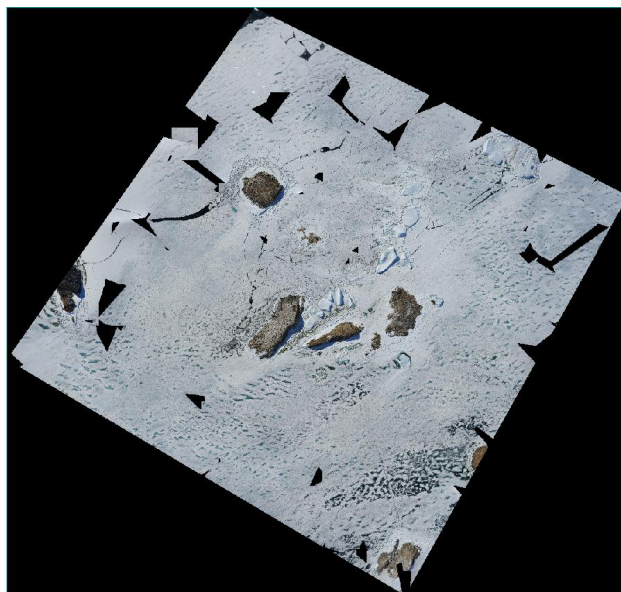


图 3-3 中山站附近企鹅岛影像资料

Figure 3-3 Image data of Penguin Island near Zhongshan Station

3.2 数据预处理

本节主要对本论文的数据预处理过程中使用的方法进行介绍，并对使用对应方法的原因和实现方式进行介绍。

3.2.1 图像切割

在准备算法模型需要的数据集时，由于企鹅栖息地岛屿影像图数据量巨大，图像中包含信息过多，无法实现直接使用该原始图像进行算法模型的训练，因此首先需要将原始图像进行切割处理。本次实验首先进行对阿德雷岛的航拍影像的图像切割，在读取阿德雷岛航拍影像后，先将该岛中企鹅密集的区域图像通过目视解译辨认出来，并将该部分区域进行提取，通过 python 脚本将原始企鹅栖息地岛屿图像进行随机切割处理，通过对算法模型和训练过程的综合考虑，避免使切割图像太小导致训练时间过长，同时也为了避免在切割图像过大导致检测精度较低，所以最后决定使切割得到的图像大小为 600×600 像素，以此作为训练算法模型的数据集图像尺寸大小。

接下来又将阿德雷岛中最大的一部分企鹅密集区目视解译出来并进行提取，接着对该区域图像从图像左上角开始进行顺序切割，通过切割时为每一张切割得到的图像保留横坐标和纵坐标的数据。本次同样使切割得到的图像大小为 $600 \times$

600 像素，以此作为训练算法模型的部分测试集。

3.2.2 图像批量命名

在对阿德雷岛图像进行切割时，根据顺序切割逻辑和图像拼接的需求，需要对切割后的图像使用具体规则来命名，使切割后的图像可以按照命名规则进行图像拼接，同时也方便分割小图被检测出企鹅后进行图像标记和图像查找。图像批量命名方法为在对图像进行顺序切割的过程中设置坐标值，使用 python 脚本实现按照坐标顺序为切割后的图像批量命名，切割后的图片在独立文件下进行保存，在测试完成后为图像拼接做准备。

3.2.3 扩充样本数量

数据扩增要求在不对原始数据造成实际数据增加的情况下，只是通过变换等形式对原始数据进行处理，来实现创造出更多数据以供使用。数据扩增主要以增加数据量、丰富数据多样性以及提高模型的泛化能力等作为实际应用过程中的目的。在进行数据扩增时，首先要保证的是不能引入无关的数据造成无效的计算。对于不同的模型任务和应用场景，数据扩增的策略也会有很大差异，所以要根据具体应用需求制定适合的数据扩增策略。同时进行数据扩增后还要保持扩增后的标签不变。在对以上数据扩增的要求进行综合考虑后，决定采用图像旋转和随机切割原始图像这两种方法来扩充数据集的样本数量。

3.2.4 数据筛选

在训练算法模型之前，对样本数量扩充后的数据集中含有企鹅的图像进行筛选，通过之前的图像数据预处理得到 10131 张图像，通过数据筛选后，得到 2000 张符合训练要求的图像。将筛选出来的图像在独立文件下进行保存，为之后的数据标注做准备。经过以上数据预处理，得到的文件形式如图 3-4 所示。

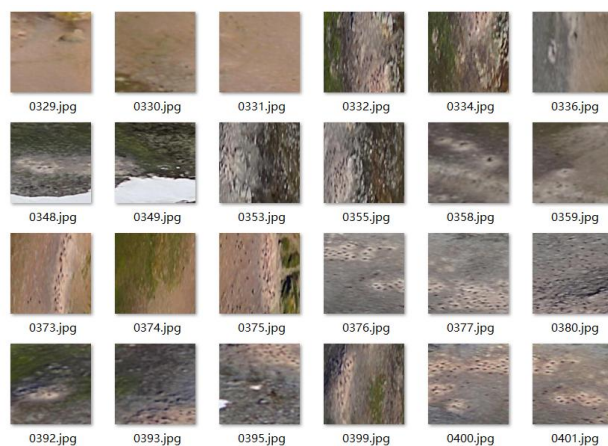


图 3-4 数据预处理后得到的部分图像

Figure 3-4 Part of the image obtained after data preprocessing

3.3 数据标注

数据标注的目的是为了创建用于训练模型所需要的数据集。在训练企鹅栖息地检测算法模型与企鹅数量检测算法模型的过程中，需要大量的带有标注的图像数据样本。并且针对不同的算法模型的训练，需要的数据集不同，因此图像标注的方式也有差异。

企鹅栖息地检测使用的 Faster RCNN 模型训练集的要求是对切割得到的图片上的企鹅区域进行标注。在寻找标注软件过程中，通过多种软件对比最终选取了图片标注工具 LabelImg，这是一款开源的图像标注工具，使 Python 语言编写并通过 Qt 框架制作可视化交互界面。LabelImg 提供三种标注模式供用户选择^[27]，“CLS”模式对整张图像进行类别标注，“DET”模式对图像中的目标进行边界框标注，“SEG”模式对图像中的目标进行实例分割标注，可以通过定位多边形边框和刷子涂抹两种方式标注目标的形状。在进行完标注后将标注信息保存为 XML 格式的文件，文件名与对应标注的图像相同，并且标注后文件的 XML 格式与 PASCAL VOC 数据集的格式相同。LabelImg 软件的使用流程如图 3-5 所示。



图 3-5 LabelImg 软件使用流程

Figure 3-5 Using process of LabelImg software

在 LabelImg 软件中的具体操作^[23]为选择 Open dir 打开存放待标注图片的文件，选择其中一张图像进行标注，使用 Create RectBox 按钮在有企鹅的区域画框，

拉伸选定了目标企鹅活动范围后点击结束画框，软件通过弹出对话框反馈用户选择标注的类别，完成类别选择后，点击 **Save** 按钮后将标注图片保存为 XML 格式文件。点击 **Next Image** 按钮后对下一张待标注图片重复之前的操作进行标注。

对于企鹅数量检测使用的 FCN 模型框架对训练集的要求是：标注切割得到的图片上的所有企鹅个体。Labelme 是语义分割的标注工具，和 LabelImg 的不同点是该软件适用于对目标进行详细的标绘，这是它和 labelImg 的最大区别，因此最终选取 Labelme 软件标注企鹅个体。Labelme 软件的使用流程如图 3-6 所示。

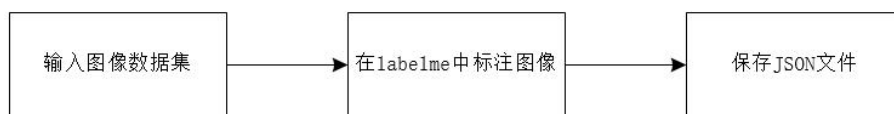


图 3-6 Labelme 软件使用流程
Figure 3-6 Using process of Labelme software

在 Labelme 软件中的具体操作与 LabelImg 相似，不同之处在于 Labelme 使用 **Create Polygons** 按钮在个体企鹅周围做标注并进行类别选择，而不再使用 **Create RectBox** 按钮在有企鹅的区域画框，因为这样做选择范围相较于目标点较大很多，不利于模型的训练。完成类别选择后，其他操作流程 LabelImg 相同，这里不再赘述。图 3-7 为使用 Labelme 进行标注的界面。

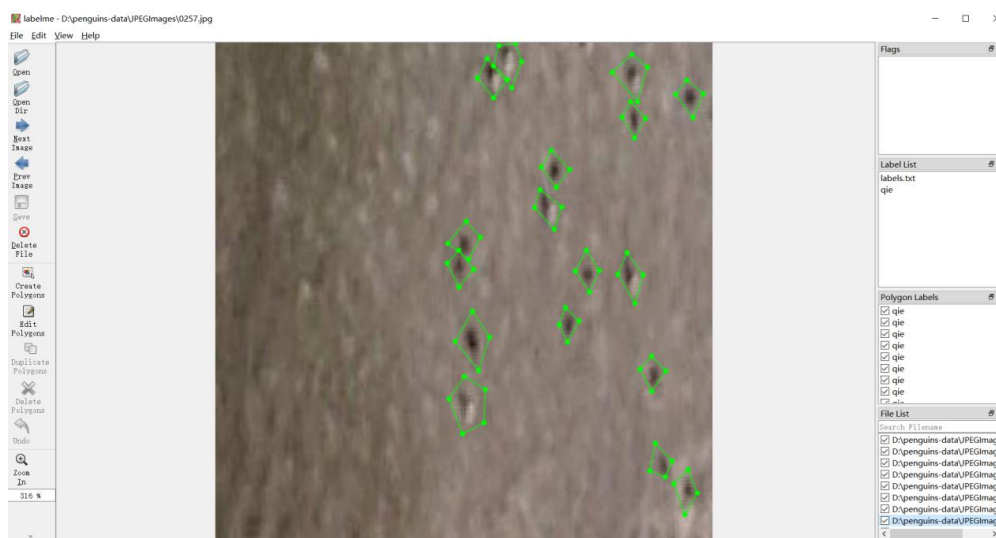


图 3-7 Labelme 标注界面图
Figure 3-7 Labeling interface of Labelme

3.4 数据转换

由于使用 Labelme 软件对图像进行标注后，保存下来的标注文件格式为 JSON 格式，我们需要将它转化为 XML 格式来作为 FCN 模型的训练数据集。具体方法为在使用标注工具标注图片目标检测框，保存生成 JSON 格式的标注文件后，使用 python 脚本对标注后的图片和标注文件名称进行批量修改，从 0000.jpg、0000.json 标号开始。待批量修改完成后，提取图片和标注文件到不同文件夹下，使用 python 脚本将读取的标注框转化为 TXT 格式，再将标注框 TXT 格式进行转化，最终得到 XML 格式。

3.5 本章小结

本章主要对数据集和图像预处理方法进行设计。通过图像切割、图像批量命名、扩充样本数量和筛选等操作来进行图像数据预处理，并通过数据标注预处理图像制作成了本文算法模型需要的数据集。

4 基于 Faster RCNN 的企鹅栖息地检测方法

企鹅栖息地检测是本系统的核心功能之一，也是系统其他核心功能的基础。本章使用 Faster RCNN 作为企鹅栖息地检测的方法，并对该方法的设计目标、改进点、具体的实验设计以及结果分析进行阐述。

4.1 设计目标

在对企鹅进行数量检测之前，首先要做到的就是明确企鹅栖息地区域的分布状况，这样开展后续企鹅数量检测工作效率会更高。同时在检测栖息地分布的情况时，我们也可以从中获取很多有价值的研究信息。

经过初步实验发现直接使用 Faster RCNN 模型应用到企鹅栖息地检测中会存在两个问题：（1）经过初次实验发现有一些小目标的栖息地并没有被检测出来，因此重新对特征点进行标注，增加小区域目标后而发现了大量的误检，在海洋、山丘等部分出现了非常多的误检区域；（2）在检测结果展示中，由于企鹅聚集分布不均匀，而且相对于整个岛屿图像来说，对切割小图进行检测得到的检测框太过微小，由此明确对于企鹅栖息地的检测结果并不是要求使用检测框对检测到的图像中的目标进行标记，而是需要更明显的标记方式来确认企鹅栖息地的分布。

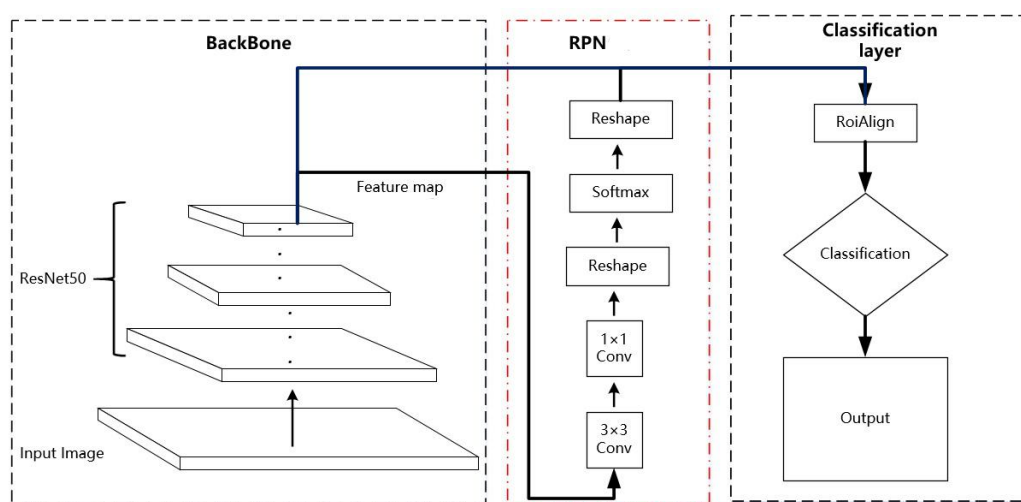


图 4-1 改进的 Faster RCNN 模型结构
Figure 4-1 Improved Faster RCNN model structure

基于上述原因，本文对 Faster RCNN 进行改进调整，首先将 ROI Align 代替 Faster RCNN 中的感兴趣区域池化来提高 Faster RCNN 对于企鹅栖息地的检测精

度，解决了实际场景中检测结果不明确、漏检小目标栖息地以及误检区域过多等问题的困扰，进一步提高了企鹅栖息地检测的准确率。然后通过取消检测框标注目标，采用对检测目标所在切割小图进行整体标注的方法，做出这样的设计后省去了计算边框回归的偏移量这一过程，提高了算法整体检测速度。整体改进算法模型结构如图 4-1 所示。

4.2 改进的 Faster RCNN

本节对 Faster RCNN 在实际使用中的两点改进方案进行介绍，首先将 ROI Align 代替 Faster RCNN 中的感兴趣区域池化来提高 Faster RCNN 对于企鹅栖息地的检测精度，并采用对企鹅所在的切割小图进行整体标注代替检测框标注目标。

4.2.1 ROI Align

根据算法的实际应用情况，我们决定将 ROI Align 代替 Faster RCNN 中的感兴趣区域池化，在此之前我们先对感兴趣区域池化层的作用和实际应用中的问题进行介绍。

传统的 CNN 网络对输入的图像尺寸要求必须是固定值，而且其输出同样是固定大小的向量或矩阵。如果不对输入图像大小做出固定要求，在提出感兴趣区域池化层之前有两种解决方法，第一种是直接在图像中截取目标所需大小的一部分图像传入网络中；第二种是直接将图像变形为所需大小后将其传输到网络中。

两种方法如图 4-2 所示。方法一对目标进行截取后破坏了图像的完整结构，方法二在图像变形后破坏了图像原始形状信息。



图 4-2 传统方法解决图像尺寸固定问题
Figure 4-2 Traditional Methods Solve the Problem of Fixed Image Size

RPN 网络是对正样本锚点采用边界回归，但以这种方式获得的候选框在大小和形状上都会产生差异，问题依旧没有得到很好地解决。因此 Faster RCNN 提出

了感兴趣区域池化层来解决此类问题。

感兴趣池化层的两个输入分别为原始的特征映射以及通过 RPN 输出的候选框，在感兴趣池化层中根据候选框的位置坐标，在特征映射中将对应区域进行池化得到固定尺寸的特征映射。因为输入的候选框是通过边界回归得到的，在这个过程中会产生浮点数，而进行池化的特征映射的大小是固定的，所以在感兴趣区域池化过程中会产生两次整数化，如图 4-3 所示。第一次是对候选框的边界处理为整数点坐标值。第二次是将整数化之后的边界区域平均分割成更小区域单元，并对每一个单元的边界再进行整数化。经过两次整数化以后，候选框的位置和开始时的对应的候选框位置相比，发生了一定的变动，很有可能影响到检测准确性，该情况被称为“不匹配问题（misalignment）”。

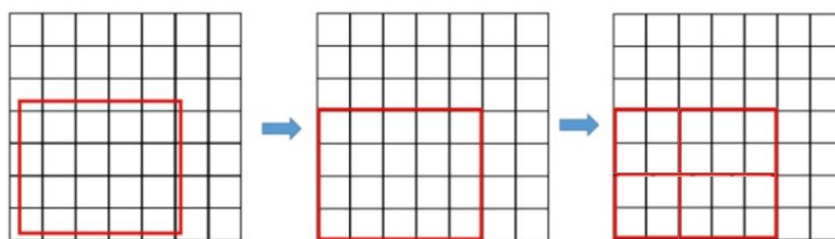


图 4-3 Faster RCNN 两次整数化过程
Figure 4-3 Two integerization process of Faster RCNN

这里用一个比较直观的例子来对过两次整数化的影响进行说明，将输入尺寸为 800×800 的图像到 Faster RCNN 框架进行检测，图像上的目标被 665×665 大小的矩形框框住。在经过特征提取后，特征映射进行缩放到原图的 $1/32$ 大小，此时原图边框为 25×25 ，而目标周围矩形框为 20.78×20.78 。接着感兴趣区域池化层将图像大小和矩形框大小进行整数化处理，图像大小仍然是 25×25 ，但是目标矩形框大小变成了 20×20 。接下来对目标矩形框进行 7×7 的特征池化处理，目标矩形框被划分为 7×7 的小矩形框，每个小矩形框的大小为 2.86×2.86 ，此时感兴趣区域池化层又小矩形框的大小进行整数化处理，得到 2×2 大小的小矩形框，此时和未经整数化处理的小矩形框相比，相差 0.8 个像素值，恢复到原图大小就会相差接近 30 个像素值，已经出现了较明显的偏差。由于企鹅有大的栖息地分布区域也有很多小型栖息地分布区域，感兴趣区域池化层产生的像素偏差肯定会影响企鹅栖息地检测的精度。

双线性插值法^[28]是为了解决在对候选框按照要求大小进行等分后，小矩形框的定点无法落到图像真实的像素点的问题。解决方法是通过每个等分小矩形中再取四个点，对每个点距离他最近的四个真实像素点的值进行加权从而得到该点的值。如图 4-4 所示。

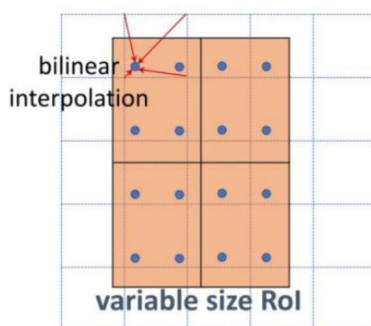


图 4-4 双线性插值示意图

Figure 4-4 Schematic of Bilinear Interpolation

ROI Align 取消了两次整数化操作，利用以上介绍的双线性插值法获取坐标为浮点数的像素点上的图像数值。ROI Align 对候选区域进行遍历后，缩放图像的过程中保留浮点数而不是进行整数化操作，在对候选区域进行更小单元的切分时同样不做整数化操作而是保留浮点数，并使用双线性插值法对每个切分的小单元进行最大池化操作获得该像素值信息。

通过 ROI Align 与 ROI Pooling 的对比，我们发现 ROI Align 在检测栖息地精度上要更胜一筹，ROI Align 在遍历取样点的数量上没有 ROI Pooling 那么多，但却可以获得更好的性能，这主要归功于解决了不匹配问题，因此我们最终做出决定，将 ROI Align 代替 ROI pooling，应用在企鹅栖息地检测算法上。

4.2.2 优化展示结果

Faster RCNN 直接使用 RPN 网络生成检测框，代替之前目标检测算法使用的滑动窗口和选择性搜索方法。RPN 网络的作用不只是通过 Softmax 分类器进行正负样本锚点分类，还有一点是计算锚点的边框回归偏移量，并将结果作为输入传到 proposal 层与候选区域结合以获得精确的候选框。这样的结构设计使算法在获得更加精确的候选框的同时，也为算法模型带来一定的计算量。

在检测结果展示中，由于企鹅聚集分布不均匀，而且相对于整个岛屿图像来说，对切割小图进行检测得到的检测框太过微小。所以对于企鹅栖息地的检测结果并不是要求使用检测框对检测到的图像中的目标进行标记，而是需要更明显的标记方式来确认企鹅栖息地的分布。如图 4-5 所示，在检测结果展示效果上呈现出来的检测框杂乱且不易辨认。



图 4-5 检测结果展示

Figure 4-5 Enlarged view of test results

由于对整张岛屿原图来说检测框过于微小，对于检测结果的查看也非常费力，所以我们取消检测框标注目标，通过对检测目标所在切割小图进行整体标注，做出这样的设计后省去了计算边界回归偏移量这一过程，间接提高了该算法模型的整体检测速度。

4.3 实验设计与结果分析

本节对该章节算法模型的实验设计和实验结果分析做出介绍，主要从实验数据集和实验配置、评价指标和实验结果与分析三个方面进行介绍。

4.3.1 实验数据集和实验配置

本次实验以阿德雷岛的无人机航拍高分辨率影像为基础进行实验分析，通过对阿德雷岛含有企鹅栖息地的区域图像进行随机切割为 600×600 的图像，同时设置随机数来判定是否进行对图像的旋转后，将含有企鹅栖息地的图像挑选出来，使用 LabelImg 标注工具进行特征区域的标注，并将标注好的图片制作成 PASCAL VOC 数据集格式，使用经过图像预处理得到数据集数据的 80% 作为训练数据，其余 20% 为测试数据对算法模型进行效果检验，通过使用 Resnet50 网络提取图片的特征图进行训练。

实验硬件环境的配置如下：CPU 为 Intel i7，内存为 16G，固态硬盘为 256G，机械硬盘为 1T。实验使用 Tensorflow 深度学习框架，并通过 CUDA 和 CUDNN 在训练过程和测试过程中进行并行加速计算。

Faster RCNN 损失函数定义见式 (4-1)，主要分为 RPN 的损失和 Fast RCNN

的损失，并且两部分损失都包括分类损失（Cls loss）和回归损失（Bbox regression loss）。

$$L(\{p_i\}, \{t_i\}) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*) \quad (4-1)$$

其中， i 是第 i 个目标候选框的索引， p_i 是第 i 个候选框为目标的概率， $t_i = \{t_s, t_b, t_w, t_h\}$ 作为一个向量，代表着预测候选框的坐标。 t_i^* 表示真实候选框的坐标向量。

$L_{cls}(p_i, p_i^*)$ 是 RPN 的分类损失，表示两个类别（目标正样本和非目标负样本）的对数损失函数，见式（4-2）。

$$L_{cls}(p_i, p_i^*) = -\log[p_i^* p_i + (1 - p_i^*)(1 - p_i)] \quad (4-2)$$

$L_{reg}(t, t_i^*)$ 是回归损失，损失函数定义见式（4-3）。

$$L_{reg}(t, t_i^*) = R(t_i - t_i^*) \quad (4-3)$$

其中， R 是 smooth L1 函数，该函数定义见式（4-4）。

$$\text{smooth}_{L1}(x) = \begin{cases} 0.5x^2, & |x| < 1 \\ |x| - 0.5, & |x| \geq 1 \end{cases} \quad (4-4)$$

由于模型中省去了计算边框回归的偏移量这一过程，不再需要进行回归损失计算，所以我们将任务综合损失函数进行修改，得到的公式见式（4-5）。

$$L(p_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) \quad (4-5)$$

训练模型时，通过随机梯度下降法，并将初始学习率设置为 $1e-4$ ，批量大小设置为 32，使用 Resnet 50 网络提取图片的特征映射进行训练，一共训练了 200 轮次（epoch），每完成一轮迭代后，通过观察训练的损失是否收敛，当发现不再收敛时表明达到目标，然后保存当前模型。

4.3.2 评价指标

在进行评估企鹅栖息地检测方法的检测效果时，实验使用误检率 FD（False detection）、漏检率 MD（Missed detection）、精确率 P（Precision）、召回率 R（Recall）和 F1 值（综合评价指标）来对该方法进行评价，其中误检率的定义为

不含有企鹅的图片被检测成含有企鹅的图片，占有所有检测结果显示为含有企鹅图片的比例；漏检率的定义为含有企鹅的图片被检测成不含企鹅的图片，占有所有含有企鹅图片的比例。计算公式见下式。

$$FD = \frac{FP}{TP + FP} \quad (4-6)$$

$$MD = \frac{FN}{TP + FN} \quad (4-7)$$

$$P = \frac{TP}{TP + FP} \quad (4-8)$$

$$R = \frac{TP}{TP + FN} \quad (4-9)$$

$$F = \frac{2 * P * R}{(P + R)} \quad (4-10)$$

其中 TP 表示将含有企鹅的图像预测为含有企鹅的图像，即真正类；FP 表示将不含有企鹅的图像预测为含有企鹅的图像，即假正类；FN 表示将含有企鹅的图像预测为不含有企鹅的图像，即假负类；F 是精确率与召回率的综合评价指标，可以通过该式调和精确率和召回率之间的矛盾。

4.3.3 实验结果与分析

为验证本文提出的企鹅栖息地检测方法的效果，本文在相同数据集的基础上，对改进前的 Faster RCNN 网络和结合本文实际应用改进后的 Faster RCNN 网络做出检测性能对比实验。

方案 1 的数据为使用改进前的 Faster RCNN 对企鹅栖息地进行检测得到的实验结果；方案 2 的数据为使用 ROI Align 代替感兴趣区域池化层的改进 Faster RCNN 对企鹅栖息地进行检测得到的实验结果；方案 3 的数据为使用 ROI Align 代替感兴趣区域池化层，并省去计算边框回归偏移量过程后的改进 Faster RCNN，对企鹅栖息地进行检测得到的实验结果。各方案的实验结果如表 4-1 所示。

通过方案 1 和方案 2 的数据对比，误检率下降了 3.41%，漏检率下降了 2.6%，对应精确率提升了 3.41%，召回率提升了 2.6%，综合评价指标提升了 3.02%，以上数据说明使用 ROI Align 代替感兴趣区域池化层的改进 Faster RCNN 对图像误检和漏检情况有一定到优化作用，使模型可以检测到更多的小目标企鹅区域并且保证发生尽可能少的误检情况，提升了整体检测的准确性。

表 4-1 不同栖息地检测方案的性能对比结果
Table 4-1 Performance comparison results of different habitat detection schemes

方案	误检率 (%)	漏检率 (%)	精确率 (%)	召回率 (%)	F1 (%)
1	11.1	9.09	88.89	90.9	89.88
2	7.69	6.49	92.3	93.5	92.9
3	7.69	6.49	92.3	93.5	92.9

通过方案 2 和方案 3 的数据对比, 两种方案的误检率、漏检率、精确率、召回率和综合评价指标都没有发生变动, 以上数据对比结果说明省去计算边框回归偏移量过程后的 Faster RCNN 的检测准确度并未发生明显变动, 但是在整体的检测速度上有一定的提升, 综合评价使用 ROI Align 代替感兴趣区域池化层, 并省去计算边框回归偏移量过程后的 Faster RCNN 的检测能力相比前两种方案更好一些。

通过不修改检测框的 Faster RCNN 进行企鹅栖息地检测得到的效果图如图 4-5 所示。我们发现图中的检测框难以辨认, 通过不展现检测框, 而是判断如果有目标存在直接对该目标图片进行强调得到更易分辨的结果。强调目标图片的方式如图 4-6 所示。图片为包含企鹅的原图切割图片, 通过企鹅栖息地检测算法对该图片进行正样本检测后, 对该图片添加红色边框以标识该图片为检测出包含企鹅的图片。



图 4-6 强调检测出的企鹅图片
Figure 4-6 Emphasize the detected penguin picture

4.4 本章小结

为了对岛屿图像中的企鹅栖息地进行精确检测, 本文使用 Faster RCNN 作为企鹅栖息地检测的方法, 结合强调检测图像代替展示检测框并用 ROI Align 代替感兴趣区域池化层用的 Faster RCNN 企鹅栖息地检测模型, 通过实验结果表明该方

法有效提升了模型对企鹅栖息地检测的精确度，并很大程度上解决了检测过程中漏检和误检的情况，并在对检测后的结果展示进行了优化，取消检测框展示的同时，也提高了算法模型整体的检测速度。

5 基于 FCN 的企鹅数量检测方法

本章利用 FCN 的特征提取能力,并结合企鹅数据集特点,对企鹅图像进行分割检测,并实现企鹅数量检测的统计。

5.1 设计目标

企鹅数量的检测是指标型数据,代表着企鹅生存状况的具体情况。通过对企鹅数量的研究,明确不同栖息地企鹅生存状况如何,以及企鹅整体生存情况。通过对同一组企鹅岛屿图像中不同栖息地数量的对比,以及不同时间对应栖息地的数量对比,都可以获得有价值的信息。

在企鹅数量识别的过程中,我们选用全卷积神经网络(FCN),FCN 是一种端到端的图像语义分割模型^[26],相较于 CNN,FCN 的主要特点如下:

(1) FCN 将 CNN 网络中后三层全连接层替换成三层卷积层,这样 FCN 实现了全卷积化,可以无限制输入图像的大小。FCN 在 CNN 的基础上将对图像级别的分类转化为对图片像素级别的分类。

(2) 使用上采样操作恢复经过不同卷积与池化得到的特征图的尺寸,使输出图像的尺寸与输入图像尺寸保持一致。

(3) 使用跳跃结构将高低层图像特征进行融合。通过这种跨层连接的结构,将不同层次的网络深层高级语义特征与浅层低级语义特征进行结合,达到对图像进行精准分割的目的。

CNN 会将原来二维矩阵的图片压扁成一维,最后通过训练输出一个标量,就是输入图像的分类标签。这一处理过程丢失了输入原始图像的空间信息,而企鹅数量检测需要的输出是个二维分割图,然后对图像的分割结果进行统计得到数量。由于 FCN 能够对图像中的特定目标进行提取,故选择 FCN 模型作为企鹅数量检测模型。

本文的企鹅数量检测模型结合了卷积化、采用反卷积作为上采样、跳跃结构等方法,其中全卷积化可以实现模型的端到端学习,采用反卷积作为上采样将低分辨率的特征图转换为高分辨率的特征图,跳跃结构对多种特征进行融合,提升模型的分割精度。

5.2 数据与方法

本节主要对基于 FCN 的企鹅数量检测模型所需要的数据集进行介绍，并对提出的 FCN 算法模型结合实际应用进行具体介绍。

5.2.1 数据集描述

在第三章中，我们对训练算法模型所需要的数据集进行了介绍，本文将用于企鹅数量检测的数据集分为训练集和测试集两部分，比例约为 8:2，其中训练集用于基于 FCN 的企鹅数量检测模型的训练，测试集用于检验该模型训练的效果，同时通过图像旋转和随机切割原始图像这两种方法来扩充样本数量以提高模型的泛化性能。

5.2.2 企鹅检测与数量统计方法

CNN 网络通过在其卷积层之后接上全连接层的方式将卷积层产生的特征映射处理成固定长度的特征向量，来得到整个输入图像的数据描述。该数值描述代表着输入图像的分类概率，是对图像级别的分类。

虽然卷积层和全连接层都是使用通过 CNN 进行训练得到的权值和偏置，但是在卷积层中的卷积核对权值和偏置的范围进行限制，所以卷积和全连接的计算概念和过程还是不同的。常用的上采样方法包括插值、上池化和反卷积。插值和上池化是不需要参数学习的，而反卷积又称转置卷积，是可以进行参数学习的。为了提高企鹅图像的分割精度，本文采用反卷积作为上采样的方式。如图 5-1 所示为卷积和反卷积的运算方式示意图。

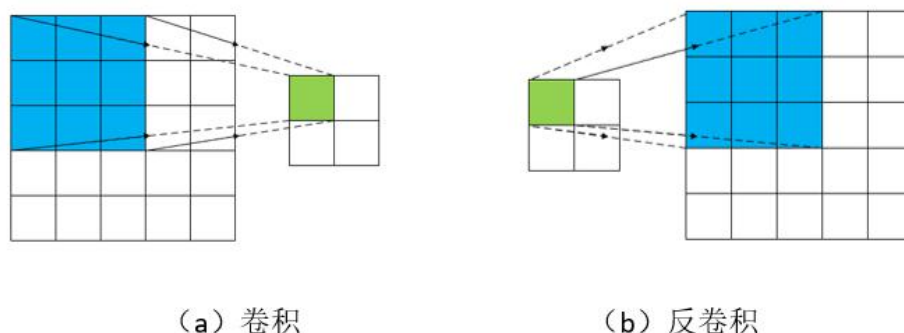


图 5-1 卷积和反卷积运算方式示意图

Figure 5-1 Schematic diagram of convolution and deconvolution operation methods

反卷积和卷积从本质逻辑上进行分析对比，我们可以发现二者是比较相似的，它们都是通过乘法和加法的运算来建立特征映射和原图像之间的联系，但是二者的对其建立联系的传播方向却是相反的。卷积是将特征图像进行缩放，而反卷积是将缩放的特征图像进行还原。而且反卷积向前传播的计算方式要高效很多，是因为其计算资源可以实现共享。反卷积过程输入与输出之间的对应关系见式(5-1)、式(5-2)，假设输入大小为 $W \times H$ ，输出大小为 $w \times h$ 。

$$w = (W - 1) \times \text{Stride} + k - 2\text{padding} \quad (5-1)$$

$$h = (H - 1) \times \text{Stride} + k - 2\text{padding} \quad (5-2)$$

其中， Stride 为卷积核滑动步长， padding 为卷积时像素间补零宽度，卷积核尺寸为 $k \times k$ 。通过反卷积对特征图像进行处理后，会产生 8 倍、16 倍、32 倍的上采样结果。

在对输入图像进行第一次卷积和池化后，得到的图像缩小为原图像大小的 1/2；在此基础上对得到的图像进行第二次卷积和池化，此时得到的图像缩小为之前得到图像的 1/2，也就是原图像尺寸的 1/4；同样的操作对上一次得到的图像进行第三次、第四次和第五次的卷积与池化处理，分别得到的图像尺寸为原图尺寸的 1/8，1/16 和 1/32。通过第六层和第七层卷积层对得到的缩小为原图 1/32 尺寸大小的图像进行卷积处理，得到热图。

通过多层卷积层的处理，我们得到的有原图 1/32 尺寸大小的热图和分别为原图 1/16、1/8、1/4、1/2 尺寸大小的多个特征图。在对热图进行上采样操作后，还原的图像特征只包含第五层卷积层处理的特征，得到 FCN-32S。由于受精度的影响，此时得到的图像无法准确还原输入图像的特征。因此 FCN 中设计了跳跃结构，将在此时对上采样处理的结果进行前向迭代，把第四层卷积层处理的特征与上一次上采样操作后得到的特征图像进行反卷积细节补充，得到 FCN-16S；把第三层卷积层处理的特征与上一次上采样操作后得到的特征图像进行反卷积细节补充，得到 FCN-8S；把第二层卷积层处理的特征与上一次上采样操作后得到的特征图像进行反卷积细节补充，得到 FCN-4S；最后把第一层卷积层处理的特征与上一次上采样操作后得到的特征图像进行反卷积细节补充，得到 FCN-2S。还原的图像精准度也越来越高。

通过基于 FCN 的企鹅数量检测模型对包含企鹅的图片进行检测，将企鹅阴影像元从图像中提取出来进行加和计算，得到该检测图像包含的企鹅像元的总数量。由于模型在训练中得到单只企鹅的平均阴影像元数量，将图片提取的企鹅阴影像元总数量对单只企鹅的平均阴影像元数量进行除法计算并向上取整，得到该检测

图片包含的企鹅数量。

5.3 实验设计与结果分析

本节对实验设计进行介绍，并将实验流程和最终得到的实验结果数据进行展示与分析，明确实验方法的可行性。

5.3.1 模型训练过程

本文采用开源的 Imagenet 数据集训练的 VGG 权重构建五种全卷积网络，并对每个网络分别训练 150 轮次。在五种模型全部训练完成后，将该数据集训练的参数作为企鹅栖息地图像数据集训练的初始值，最后将五种模型分别训练 150 轮次，得到最终的分割结果。

5.3.2 评价指标

在进行评估企鹅数量检测方法的检测效果时，实验使用精确率 P (Precision)、R 召回率(Recall)和 F1 值(综合评价指标)来衡量该方法的有效性，计算公式如下。

$$P = \frac{TP}{TP + FP} \quad (5-3)$$

$$R = \frac{TP}{TP + FN} \quad (5-4)$$

$$F = \frac{2 * P * R}{(P + R)} \quad (5-5)$$

其中 TP 表示企鹅阴影像元预测结果正确，即真正类；FP 表示将非企鹅阴影像元预测为企鹅阴影像元，即假正类；FN 表示将企鹅阴影像元预测为非企鹅阴影像元，即假负类；F 是精确率和召回率的综合评价指标，可以通过该式调和精确率和召回率之间的矛盾。

5.3.3 实验结果与分析

本文从模型的企鹅数量检测准确度和分割速度两方面分别进行评价，对该模型的整体性能进行评估，不仅要考虑模型的数量检测准确度，分割速度也是展示

模型性能的重要指标。分割的速度越快代表模型的复杂度越低，模型在实际应用时对硬件环境的要求越低，所以能耗也相对较小。本文使用 20 张测试集图像，分别测试了模型的分割精度和分割速度。

企鹅数量检测准确度采用精确率 P (Precision)、R 召回率(Recall)和 F1 值(综合评价指标)来衡量该方法的有效性。分割准确度的三个指标数值越大，说明模型的分割准确度越高，性能越好。实验结果如表 5-1 所示。

表 5-1 不同方法的分割准确度对比结果
Table 5-1 Comparison results of segmentation accuracy of different methods

方法	精确率 (%)	召回率 (%)	F1 (%)
FCN-32s	77.42	79.32	78.36
FCN-16s	78.67	80.47	79.56
FCN-8s	82.35	88.63	85.37
FCN-4s	82.86	89.35	85.98
FCN-2s	83.22	90.14	86.54

从表中可以看出，从以上指标可以看出融合的特征越多的模型，其分割精度越高，其中 FCN-2s 的分割精度最高，由此可以看出融合多种特征有利于提高企鹅图像的分割精度。

但是从数据上看，FCN-8s 与 FCN-4s、FCN-2s 的数量检测准确度区别相差并不大，但是 FCN-8s 与 FCN-16s、FCN-32s 的数量检测准确度差距远高于 FCN-8s 与 FCN-4s、FCN-2s 的数量检测准确度。此时我们进行分割速度的对比分析。

表 5-2 不同方法的分割时间对比结果
Table 5-2 Comparison results of segmentation time of different methods

方法	单张图片分割时间 (s)
FCN-32s	1.33
FCN-16s	1.46
FCN-8s	1.96
FCN-4s	2.53
FCN-2s	4.41

分割速度的测试结果如表 5-2 所示。从表中可以看出五种全卷积网络中，FCN-32s 的分割速度最快，这是因为在上采样阶段该模型的运算量最小，随着跳跃结构融合的层数逐渐增多，模型运算量同时也相应地逐步增大，最终导致模型的分割速度逐渐降低，FCN-2s 的单张图片分割时间比 FCN-8s 的两倍时长还要多一

些。结合模型的数量检测准确度和分割速度两方面的综合考虑, 最终选择使用 FCN-8s 作为企鹅数量检测的模型。

5.4 本章小结

本章详细阐述了企鹅数量检测的方法, 主要包括:设计的目标、方法的设计和材料、实验训练的流程、统计企鹅数量的方法以及得到训练结果后, 从模型的分割精度和分割速度对模型性能进行了评价。本章结合企鹅栖息地图像数据集的特点, 采用合理的训练方法在企鹅数量检测方面达到了较好的分割效果

6 企鹅栖息地与数量检测系统的设计与实现

本章节主要对企鹅栖息地与数量检测系统进行需求分析、系统概要设计和系统各功能模块设计与实现的阐述，并对系统测试结果进行展示。该系统具有上传企鹅岛屿图像、图像预处理、企鹅栖息地检测、企鹅数量检测和检测对比等功能。本系统采用 C/S 模式作为整体架构^[29]，系统前端使用 HTML5, CSS3 及 JavaScript 技术实现，后端开发语言使用 Python，开发框架采用 Django，将本文改进的 Faster RCNN 和 FCN 算法嵌入到 Django 中，对企鹅岛屿图像进行检测。开发的 Web 应用提供给用户操作界面，方便用户使用该系统进行上传、检测和下载等。

6.1 系统需求分析

在进行系统设计之前，首先要明确的是系统的设计目标和核心功能，然后才能有针对性的对系统进行需求分析，并在此基础上对系统整体功能和性能进行分析和总结。

6.1.1 需求分析综述

过去对于企鹅栖息地和企鹅数量的获取大都通过借助人目视解译的方式从中高分辨率遥感影像中，将企鹅粪便区域从影像中解译出来，在这基础上通过建立企鹅粪便区面积与企鹅数量回归关系的方式来获取企鹅数量。在企鹅登陆繁殖的季节，遥感影像的获取与解译容易受到云层的影响，而且在对企鹅粪便区面积进行提取的过程中，因为存在提取误差以及企鹅密度分布不均等问题的存在，会对该方式获取企鹅数量信息的准确性有较大影响。

近年在高分辨率卫星影像和航空摄影遥感技术加速发展的时代背景下，对南极企鹅研究的制图精度得到明显提高，利用企鹅像元直接研究企鹅种群分布成为可能。无人机由于其体积小、操作方便和分辨率高等优点，近年来常用于南极企鹅种群监测，打破了传统遥感方法的限制。

为克服上述企鹅栖息地与数量信息获取方式存在的弊端，同时也为了降低降低统计过程中所需消耗的人力成本并提高统计效率，本系统通过使用基于 Faster RCNN 的企鹅栖息地检测方法对南极企鹅岛企鹅栖息地进行检测，使用基于 FCN 的企鹅数量检测方法对栖息地中企鹅进行检测并统计企鹅数量。同时系统提供检测对比展示功能对栖息地检测结果信息进行比较，直观地展示企鹅栖息地的变动

情况，为企鹅与南极生态的环境保护工作提供重要的数据支持。

6.1.2 功能性需求

系统实现的前提必不可少的一部分就是详细的功能需求分析，在明确系统各模块所包含的功能点的情况下，对各功能所需要达到的效果目标和程度给出详细的说明后，才能进行后续的开发工作。由之前几章的工作可知，本系统设计利用基于目标检测技术对南极企鹅岛屿企鹅栖息地进行检测，确定企鹅栖息地的位置分布，使用语义分割方法对栖息地中企鹅进行检测并对检测得到的数量进行统计，以获取上传原始图像内企鹅的数量信息。通过系统提供的检测结果信息，以及检测对比功能对检测结果信息进行比较，可以准确、高效的提供企鹅研究的相应数据，为企鹅的保护和南极生态的环境保护提供数据支持。系统的用例图如图 6-1 所示。

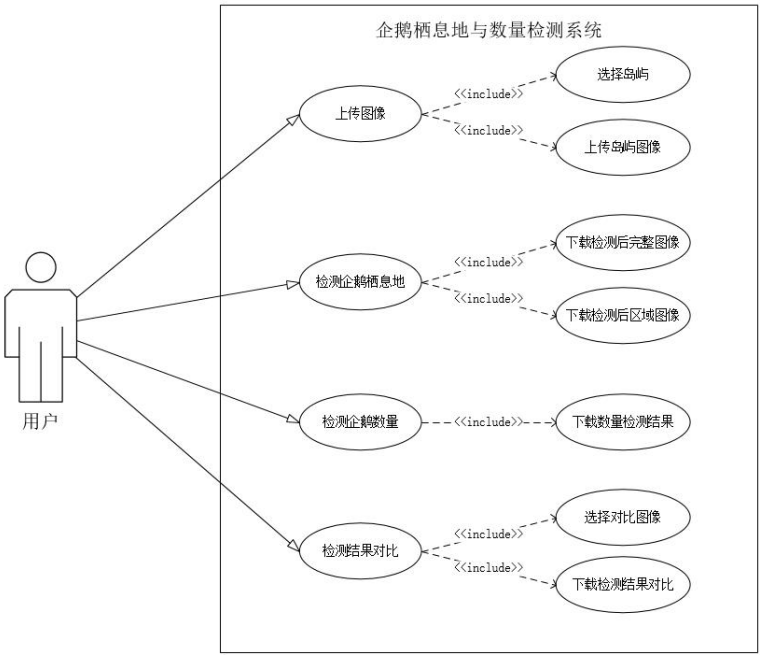


图 6-1 系统用例图
Figure 6-1 System use case

系统目标功能有四点，分别为图像上传功能、企鹅栖息地检测功能、企鹅栖息地检测功能和检测对比功能，各功能介绍如下。

(1) 图像上传功能

本系统进行企鹅栖息地与数量检测的相关功能是在南极长城站附近阿德雷岛、拜尔斯半岛和中山站附近企鹅岛这三组数据基础上进行设计与开发的，因为

三组图像特征及拍摄条件的差异,我们分别对每组数据进行检测功能的模型训练。因此在用户上传图像时需要选定图像对应的岛屿后再进行上传。上传成功后系统对原始图像进行切割,为企鹅栖息地检测和企鹅数量检测提供数据准备。

(2) 企鹅栖息地检测功能

在对企鹅进行数量研究和迁徙习性研究之前,首先要做到的就是明确企鹅栖息地区域的分布状况,只有在充分了解并分析企鹅栖息地的分布情况后,才能开展后续的研究工作。并且在研究栖息地分布的情况时,也可以从中获得很多有价值的信息。

(3) 企鹅数量检测功能

企鹅数量是指标型数据,代表着企鹅生存状况的具体情况。通过对企鹅数量的研究,明确企鹅生存状况和整个岛屿企鹅数量的分布状况,可以获取更多的研究信息。

(4) 检测对比功能

通过对企鹅栖息地分布状况和变动情况的分析,对企鹅生存状况和栖息地迁徙的研究工作都有很大的帮助,系统通过提供检测对比功能对同一岛屿不同时期的栖息地检测结果进行比较,来获取更多的研究信息。

6.1.3 非功能性需求

除了对功能性需求进行分析以外,我们还需要对一些非功能性需求进行补充分析。非功能性需求是在某些条件下判断系统运行情形或者其特性,而不是针对系统特定功能定制的系统需求^[30],主要包括了准确性需求、稳定性需求、易用性需求和可靠性需求等四个方面。这些非功能性需求的实现不仅对系统整体质量的提高有积极影响,还对功能性需求的实用价值提高有着很大的影响。因此我们将从这四个方面对该系统的非功能性需求进行分析与定义。

(1) 准确性需求

因为本系统的主要功能是进行企鹅栖息地的检测和企鹅数量的检测,所以系统首先要保证的就是检测需要有较高的准确性,包括栖息地的检测准确性和企鹅数量的检测准确性,该需求是本系统的核心价值。

(2) 稳定性需求

系统的稳定性需求是系统价值的重要体现,对于一个不稳定的系统来说,其使用价值和实用价值必定会大打折扣。使用系统的意义一是有完成特定任务的需求,二是为了提高工作效率。如果系统不稳定,即使能够完成任务,也势必会大幅削减工作效率,甚至面对更严重的问题出现时,存在无法完成任务的风险。保

证系统的稳定性，就是要保证在系统出现问题时，系统数据的准确性和完整性不受影响。

(3) 易用性需求

本系统的操作应该简单，具备易用的人机交互界面，方便用户进行操作使用。

(4) 可靠性需求

由于我们的图像上传功能以及检测功能一般会持续较长时间，因此本系统需要在使用期间应当能够长时间运行，以此来保障用户行为不被中断。当检测期间被强行中止时，系统可记录当前检测日志，并在功能恢复后可以根据日志记录继续进行检测，使系统减少无效计算。

6.2 系统概要设计

经过对企鹅栖息地与检测系统进行需求分析之后，我们对该系统的功能性需求和非功能性需求已经有了深刻理解，接下来按照需求分析去完成本系统的概要设计部分。本节进行概要设计的目标是将系统按照需求进行具体的模块划分，明确各模块之间的层次关系并确定模块之间作用关系，再根据分析得到的系统功能模块的划分情况，建立系统的技术架构，为系统提供具体的解决方案。进行概要设计的目的是完成系统逻辑模型的搭建，为此后的系统详细设计与实现提供了分析基础。

在本系统的设计过程中，我们将系统划分为四大核心模块，分别是图像上传模块、企鹅栖息地检测模块、企鹅数量检测模块和检测对比模块。本系统功能结构图如图 6-2 所示。

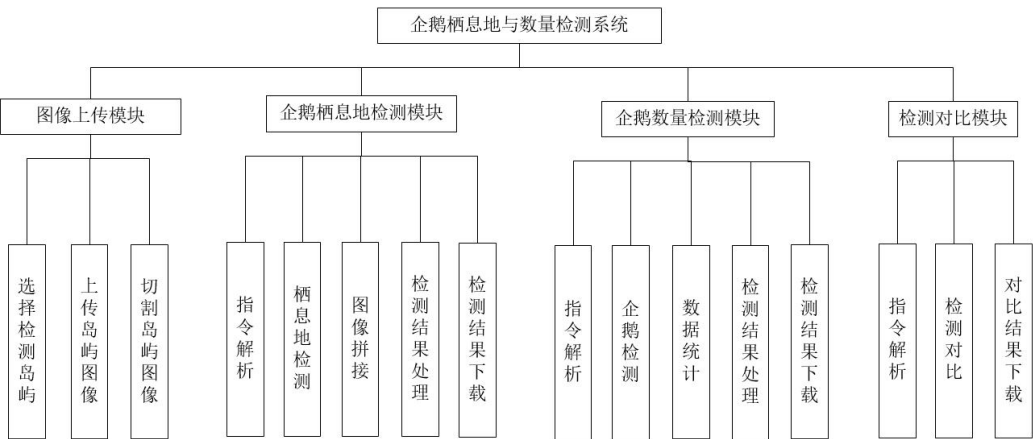


图 6-2 系统功能结构图
Figure 6-2 System functional structure

各个模块以及对应功能的概要介绍如下：

（1）图像上传模块

图像上传模块包含三个主要功能，分别为选择目标检测岛屿功能、上传岛屿图像功能和切割岛屿图像功能。选择检测岛屿功能是需要用户对准备上传岛屿的类别在系统中进行选择确认，才可以进行上传岛屿图像功能。此功能需要用户明确原始图像是阿德雷岛、拜尔斯半岛和中山站附近企鹅岛这三座岛屿中哪一座的航拍图像，进行选择确认后才可以进行上传岛屿图像功能。因为三组图像特征及拍摄条件的差异，我们分别对每组数据进行检测功能的模型训练，如果选择的岛屿与上传的岛屿图像不匹配，系统无法保证检测准确度。进行上传岛屿图像功能操作成功后，系统对原始图像进行切割，切割的同时对切割后得到的小图进行规则化命名，为企鹅栖息地检测和企鹅数量检测提供数据准备。

（2）企鹅栖息地检测模块

企鹅栖息地检测模块包含五个部分，分别为指令解析、栖息地检测、图像拼接、检测结果处理和检测结果下载。在系统完成原始岛屿图像切割后，需要用户选择确认进行企鹅栖息地检测。在用户发起该确认操作后，系统进行指令解析，完成接收前端发出的指令并开始对之前分割原始图像得到的小图进行企鹅栖息地检测。图像拼接是指检测完成后系统生成一套对应原始分割小图的标注图像，此时系统将该整套图像的全部图像进行拼接，生成含有标注信息的原始大图。除了将检测处理得到的图像进行拼接以外，系统仍然保留检测出的包含企鹅的切割小图，用户可以通过执行检测结果下载，分别选择下载未经拼接处理检测出的企鹅图像，或者拼接后含有标注信息的原始大图。

（3）企鹅数量检测模块

企鹅数量检测模块包含五个部分，分别为指令解析、企鹅检测、数量统计、检测结果处理和检测结果下载。在系统完成企鹅栖息地检测后，需要用户确认后，进行企鹅数量检测。在用户发起该确认操作后，系统进行指令解析，完成接收前端发出的指令，开始对企鹅栖息地检测出包含企鹅的图像进行企鹅检测。检测完成后，系统对每张小图的企鹅数量进行统计，并通过数据统计功能对企鹅整体数量进行统计，同时用户可以在系统查看检测到的企鹅总数量，并可以在用户选择检测结果下载后对检测得到的详细数据进行下载。

（4）检测对比模块

检测对比模块包含三个部分，分别为指令解析、检测对比、对比结果下载。该模块功能主要是对同一岛屿不同时期图像的检测结果进行对比，用户使用该模块功能之前，需要在系统中至少对同一岛屿进行两次不同航拍图像的栖息地检测。在用户发起该确认操作后，系统进行指令解析，提示用户选择需要进行检测对比

的两组图像，待用户选择确认后，系统开始进行检测对比功能。检测对比结束后，用户可以下载检测对比结果。

本系统的基本架构设计如图 6-3 所示，主要包含三个版块，分别为展示层、应用层、模型数据层。本系统采用 C/S 模式作为整体架构，其中展示层为前端系统，使用了 HTML5、CSS3 及 JavaScript 技术实现；应用层包含了系统各功能模块，后端开发语言使用 Python，开发框架采用 Django。在模型数据层中，主要包含网络模型模块以及图像检测算法模块和数据库。数据库则采用了关系型数据库 MySQL，用来存储检测后得到的数据信息。网络模型则基于 Tensorflow 深度学习框架进行搭建。

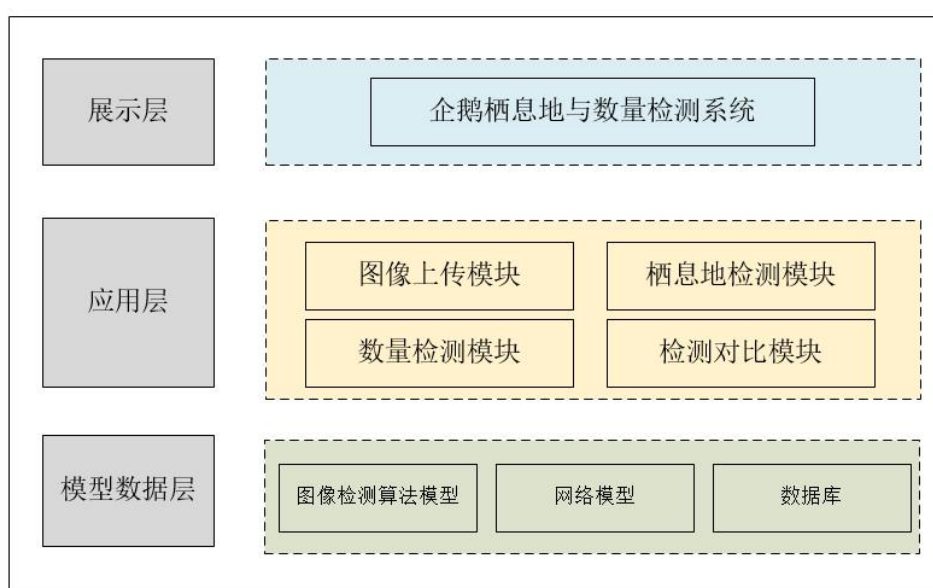


图 6-3 原型系统基本架构设计

Figure 6-3 Basic architecture design of the prototype system

6.3 系统功能模块设计与实现

本节主要对系统的图像上传模块、企鹅栖息地检测模块、企鹅数量检测模块和检测对比模块这四个模块进行详细设计，并对各模块功能的实现方法和操作逻辑进行说明。企鹅栖息地与数量检测系统前端部分通过网页的形式为企鹅栖息地与数量检测系统用户提供操作界面。系统界面按照系统功能的类型和功能所在的模块，划分为图像上传页面、主页面、栖息地检测页面、数量检测页面、检测对比页面和历史记录页面。

6.3.1 图像上传模块

图像上传模块包含三个主要功能，分别为选择检测岛屿功能、上传岛屿图像功能和切割岛屿图像功能。选择检测岛屿功能的设计是需要用户根据上传图像岛屿的类型进行确认后，进行上传岛屿图像功能。当用户在前端界面点击三个岛屿按钮中的某一个按钮时，会执行该按钮对应的方法。三个不同的方法在执行后匹配到的系统检测模型不同。

上传岛屿图像功能的设计是为了实现从本地磁盘选择岛屿图像进行上传。上传岛屿图像功能需要用户先对岛屿类型进行确认后才能进行上传图像的操作。由于从本地磁盘中获取到的图像是完整岛屿的原始图像，并不能直接通过系统的企鹅栖息地检测模块进行企鹅栖息地检测，需要先对原始图像进行切割等预处理。切割岛屿图像功能利用了 Python 工具包完成，切割的同时对切割得到的小图进行规范化命名，为企鹅栖息地检测和企鹅数量检测提供数据准备。图 6-4 为图像上传模块流程图。

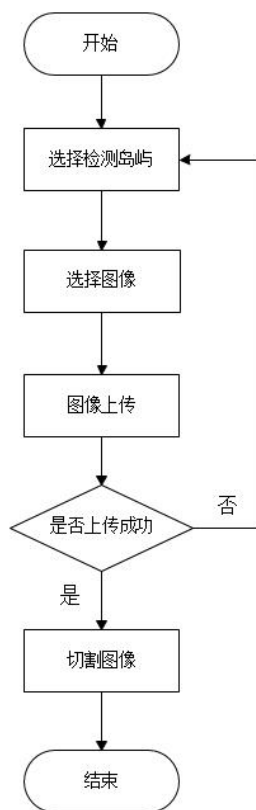


图 6-4 图像上传模块流程图

Figure 6-4 Flow chart of Image upload module

图像上传页面是用户打开系统后的初始页面,如图 6-5 所示。当前页面包含五个按钮,三个按钮为选择按钮,分别为阿德雷岛、拜尔斯半岛和中山站附近企鹅岛这三座岛屿对应的检测模型。用户需要先选中其中一个按钮后,此时选中按钮颜色变成灰色,表示按钮被选中。再点击“上传图像”按钮,按钮颜色变蓝,表示按钮被点击(系统中除了图像上传页面的三个岛屿按钮以外,其他所有按钮被选中后都会显示蓝色),进入选择本地文件页面,在该页面上选取需要上传的图像文件。上传图像到系统中,是用户在系统中使用所有功能的基础,用户上传图像完成后,才可以进入后续页面。在图像上传完成后,系统通过弹窗提示用户图像上传成功,此时进入系统主页面。当用户选取“历史记录”按钮时,页面跳转到历史记录页面。该页面向用户展示历史上传的图像信息,信息内容包括“图像名称”、“上传成功时间”、“栖息地检测完成”,通过列表形式对图像信息进行展示,每条信息的右侧有对应的选择框,当用户选中一条图像信息,点击“查看信息”后,此时也可以进入系统主界面。



图 6-5 图像上传页面

Figure 6-5 Original design of the Image upload page

主页面汇总了栖息地检测、数量检测和检测对比这三个核心功能,如图 6-6 所示。图中四个按钮分别对应“栖息地检测”、“数量检测”、“检测对比”和“重新上传图像”这四个功能。用户点击“栖息地检测”按钮后,系统将对用户上传的图像进行栖息地检测,此时系统页面中心显示“正在检测中...”的字样,用户此时可进行的操作只有取消检测过程,具体操作方式为点击检测提示中的“取消检测”按钮。系统检测完成后,通过弹窗提示用户栖息地检测成功,用户将进入栖息地检测页面,在该页面可以进行下载企鹅栖息地检测结果的操作。



图 6-6 主页面

Figure 6-6 The original design of the Main page

6.3.2 企鹅栖息地检测模块

企鹅栖息地检测模块的主要功能就是检测企鹅栖息地的位置，并对检测到的企鹅图像进行标注。用户可以对标注后的图像数据进行下载。在系统接收前端发出的指令并解析出指定的栖息地检测操作后，将图像信息输入到企鹅栖息地检测模型，调用企鹅栖息地检测算法对企鹅图像进行检测，并对检测出的含有企鹅的图像进行标注。完成检测后，对检测得到的结果进行处理。分类处理完成后将检测到包含企鹅的图片导入到企鹅数量检测算法模型中，为之后进行企鹅数量检测做准备。系统根据包含企鹅的标注图片坐标和未进行标注的不包含企鹅的图片坐标对这两类图片进行拼接，拼接得到完整的岛屿大图，此时的大图因为包含标注信息，可以直观展示企鹅栖息地分布。系统的检测信息（包含企鹅的图片名称和对应图片坐标）通过 XLWT 模块实现从 MySQL 导出数据到 EXCEL 表中，并提供一个可下载的按钮，使用户可以离线检查检测结果。图 6-7 为企鹅栖息地检测模块流程图。

在栖息地检测页面中，用户可以对栖息地检测结果进行下载，此模块包含检测完整图像下载功能和检测区域图像下载功能。如图 6-8 所示。图中三个按钮分别指示了“检测完整图像下载”、“检测区域图像下载”、“检测信息下载”和“返回上页”四个功能。在该界面下，用户可以通过点击“检测完整图像下载”按钮下载系统拼接的完整图像。在下载过程中，用户可以点击“取消下载”按钮取消对该图像的下载，点击取消操作后，系统回到栖息地检测页面。等待图像下载成功后，系统通过弹窗提示用户下载成功。用户可以通过点击“检测区域图像下载”按钮打包下载包含企鹅的切割小图。用户点击“检测结果下载”按钮下载

EXCEL 文件。

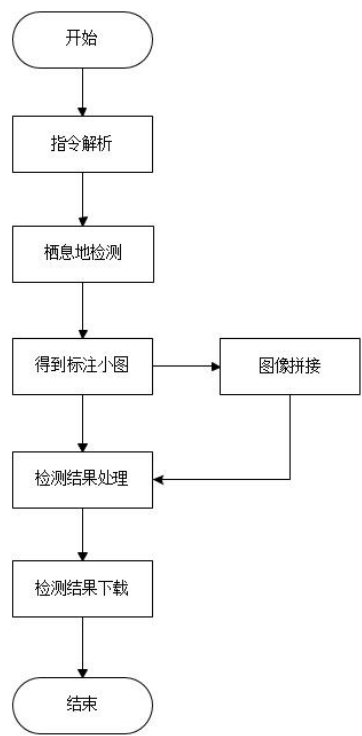


图 6-7 企鹅栖息地检测模块流程图
Figure 6-7 Flow chart of Penguin habitat detection module

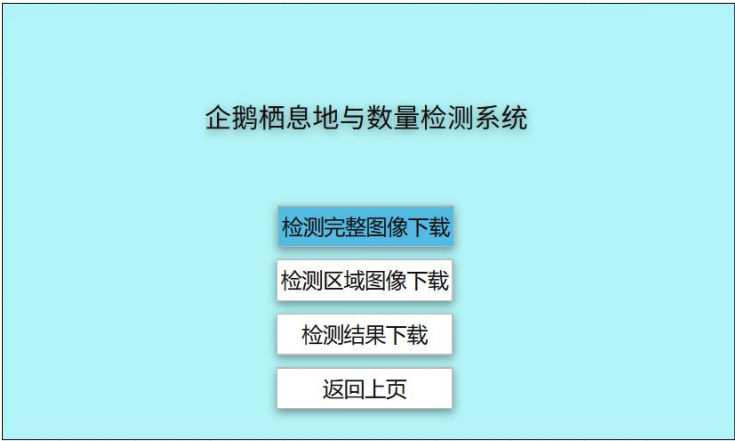


图 6-8 企鹅栖息地检测页面
Figure 6-8 Original interface design of Penguin habitat detection page

6.3.3 企鹅数量检测模块

系统接收前端发出的指令并解析出指定的企鹅数量检测操作后，将企鹅栖息地检测得到的企鹅图像输入到企鹅数量检测模型，调用检测算法对企鹅图像进行

图像分割，接着将图像分割后得到的结果进行企鹅数量统计。系统完成数量统计后，对检测得到的结果进行处理，把返回给系统的检测信息（包含企鹅的图片名称、图片坐标和该图片包含的企鹅数量）从 MySQL 中导出数据到 EXCEL 表，并提供一个可下载的按钮，从而使用户可以离线检查检测结果。图 6-9 为企鹅数量检测模块流程图。

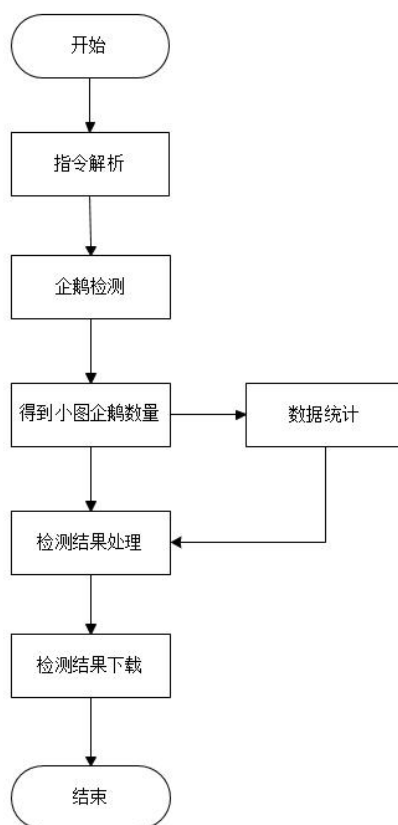


图 6-9 企鹅数量检测模块流程图

Figure 6-9 Flow chart of Penguin Number Detection Module

用户点击“数量检测”按钮后，系统将对用户上传的图像进行数量检测，此时系统页面中心显示“正在检测中...”的字样，用户此时可进行的操作只有取消检测过程。等待系统检测完成后，提示用户数量检测成功，用户进入数量检测页面，在该界面可以对企鹅数量检测结果进行下载。

企鹅数量检测页面包含展示检测企鹅总数量和数量检测结果下载功能。如图 6-10 所示。图中央对系统检测图像得到的企鹅总数量进行展示，图中两个按钮分别指示了“数量检测结果下载”和“返回上页”两个功能。在该界面下，用户通过点击“数量检测结果下载”按钮下载 EXCEL 文件，文件内容包含图片名称、图片横坐标、图片纵坐标、图片企鹅数量和企鹅总数量等信息。

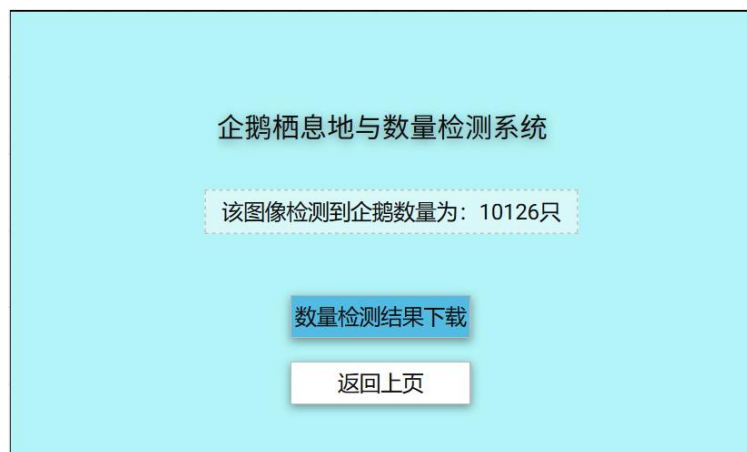


图 6-10 企鹅数量检测页面
Figure 6-10 Penguin number detection module

6.3.4 检测对比模块

在栖息地检测完成后,使用 Matplotlib 将检测后确认为企鹅图像的图像坐标数据构建一个坐标系,在坐标系中对检测到的含有企鹅图像坐标进行点集标识,来表现出整体图片上显示企鹅栖息地的分布状况。通过上传的同一岛屿的两组图像完成栖息地检测后,在检测对比模块对这两组图像数据进行栖息地检测对比,可以将坐标中的标记点集进行合并处理,通过坐标系展示可以直观发现两组企鹅栖息地分布的差异,且可以通过系统对坐标系图片进行下载。如果只选中一张图像数据进行检测对比,虽然执行了检测对比的功能,但由于只有一张图像数据进行检测对比,结果中的坐标系只有该图像的栖息地分布点集。图 6-11 为检测对比模块流程图。

用户在系统主页面点击“检测对比”按钮,首先进入历史记录页面。用户选择需要进行对比的两组同岛屿图像数据后,选中两者对应的信息选择框后,点击“开始对比”,系统将对用户选择的两组图像进行对比分析,此时系统页面中心显示“正在对比中...”的字样,用户此时可进行的操作只有取消检测对比过程。等待系统检测完成后,系统提示用户检测对比完成,用户将进入检测对比模块界面,在该界面可以对检测对比结果进行下载。

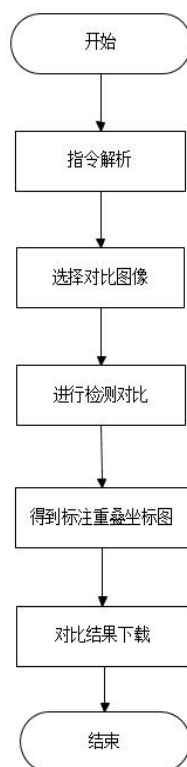


图 6-11 检测对比模块流程图

Figure 6-11 Flow chart of comparison detection module

如图 6-12 所示。图中两个按钮分别指示了“检测对比图像下载”和“返回上页”这两个功能。在该功能模块下，系统通过黑色和红色这两种颜色分别标注两组数据图像在坐标系中的点集，并通过重叠的方式将两组图像标注在同一坐标系中，黑色点集表示最近时间进行栖息地检测的图像上。在该页面下，用户可以通过点击“检测对比图像下载”按钮下载检测对比后进行标注的坐标图像。

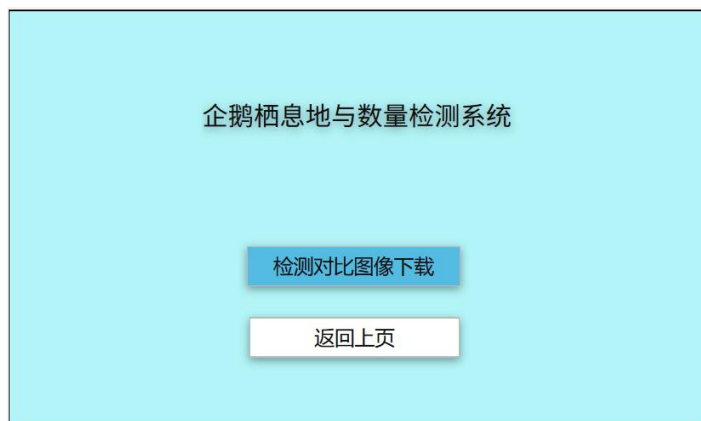


图 6-12 检测对比页面

Figure 6-12 Contrast detection page

6.4 系统测试

通过对之前工作的论述和实现，企鹅栖息地与数量检测系统已经基本实现，本节主要对企鹅栖息地与数量检测系统进行功能测试。

本系统基于 WEB 平台，系统的操作流程简易且方便，通过从本地磁盘中选择图像进行上传，接着对应检测处理的按钮进行点击，就可以实现系统对图像的检测处理过程。检测处理完成后，用户可通过点击按钮的方式实现对检测结果的下载，满足了系统易用性的需求。

系统测试时选择对整张阿德雷岛的航拍影像进行测试，成功上传图像并对检测结果进行下载，以下图片对下载的检测结果进行展示。包括企鹅栖息地检测完整图像放大图、企鹅栖息地检测区域图像下载部分结果和企鹅数量检测下载部分结果，分别如图 6-13、图 6-14 和图 6-15 所示。通过对各功能进行测试，并将测试数据与图片真实数据进行对比，功能的完整性和准确性都得到了验证。



图 6-13 企鹅栖息地检测完整图像放大图
Figure 6-13 Enlarged Image of Penguin Habitat Detection



图 6-14 企鹅栖息地检测区域图像的部分结果下载
Figure 6-14 Penguin habitat detection area image download partial results

图片名	图片横坐标	图片纵坐标	企鹅数量
1581	10	91	82
1582	10	92	26
1583	10	93	43
1595	10	105	14
1596	10	106	56
1605	10	115	45
1606	10	116	24
1607	10	117	36
1631	10	141	23
1632	10	142	31
1633	10	143	29
1634	10	144	15
1729	11	90	32
1730	11	91	77
1731	11	92	66
1738	11	99	18
1739	11	100	35
1745	11	106	31
1746	11	107	43
1876	12	88	55
1877	12	89	36
1878	12	90	25
1879	12	91	21
1885	12	97	24
1886	12	98	54
1890	12	102	38
1891	12	103	59

图 6-15 企鹅数量检测下载部分结果
Figure 6-15 Partial download results of penguin number detection

6.5 本章小结

本章介绍了企鹅栖息地与数量检测系统的设计和实现。该系统主要分为图像上传、企鹅栖息地检测、企鹅数量检测和检测对比四个模块，并从设计与实现上对各个模块进行阐述。最后进行了企鹅栖息地与数量检测系统的相关测试，通过测试结果和系统检测结果证明了本系统在企鹅栖息地与数量检测上都具有较好的效果。

7 结论

本章将对论文工作进行总结，整理企鹅栖息地与数量检测系统开发和应用中的工作内容，对系统中可以进行改进的点和不足之处进行分析和总结，从而提出对未来工作的展望。

7.1 论文总结

南极企鹅是南极地区环境变化的“生物指示剂”，研究南极典型地区企鹅栖息地分布与种群数量，将为探讨气候变化对企鹅影响及气候变化条件下企鹅物种保护提供基础数据及理论指导。本论文实现利用 Faster RCNN 作为企鹅栖息地检测的方法，对企鹅栖息地进行检测查找，使用基于 FCN 的企鹅数量检测方法对栖息地图像中的企鹅进行分割检测并统计企鹅数量，并开发出对应功能的企鹅栖息地与数量统计系统，减少人工参与企鹅栖息地检测和企鹅数量统计的过程，降低人力成本并提高检测与统计的效率，为企鹅的保护和南极生态的环境保护提供数据支持。本论文主要工作内容和结论如下：

(1) 之前对于企鹅栖息地的检测很大程度上需要依靠人工目视解译的参与。本论文实现使用 Faster RCNN 作为企鹅栖息地检测的方法，解决了检测中需要人工参与目视解译的过程，并针对 Faster RCNN 在实现企鹅栖息地检测的过程中遇到的漏检、误检问题，以及对企鹅栖息地检测后的展示结果不易辨认等问题，提出了一种改进 Faster RCNN 的企鹅栖息地检测方法。在 Faster RCNN 的基础上将 ROI Align 代替 Faster RCNN 中的 ROI Pooling 来提高 Faster RCNN 对于企鹅栖息地的检测精度，使误检率下降了 3.41%，漏检率下降了 2.6%，对应精确率提升了 3.41%，召回率提升了 2.6%，综合评价指标提升了 3.02%。并通过取消检测框标注目标，采用对检测目标所在切割小图进行整体标注的方法，使企鹅栖息地在检测后拼接得到的岛屿图像中更易辨认出来。且做出这样的设计后省去了计算边框回归偏移量的过程，间接提高了算法整体检测速度。

(2) 由于基于 Faster RCNN 的企鹅栖息地目标检测方法已经将企鹅栖息地检测出来，本论文将企鹅栖息地检测结果作为企鹅数量检测的输入数据，结合基于 FCN 的企鹅数量检测方法进行对栖息地中的企鹅进行检测并统计企鹅数量，该方法提高了企鹅检测与数量统计的效率，并保证检测过程中，精确率达到 82.35%，召回率达到 88.63%，综合评价指标达到 85.37%。

(3) 本论文将之前提出的企鹅栖息地检测方法与企鹅数量检测方法集成起

来，建立了一个企鹅栖息地与数量检测系统。通过导入需要被检测的企鹅岛屿图像到该系统，经过后端处理后得到企鹅栖息地和企鹅数量的检测结果，并可将检测结果导出查看。

7.2 工作展望

虽然企鹅栖息地与数量检测系统在企鹅栖息地检测和企鹅数量检测的功能上可以提供一定的价值，但是系统仍然有一定的优化空间，未来企鹅栖息地与数量检测系统应该可以从以下三个方面去改善和提高。

（1）虽然在基于 Faster RCNN 的企鹅栖息地目标检测方法中，将 ROI Align 代替 Faster RCNN 中的 ROI Pooling 可以实现 Faster RCNN 对于企鹅栖息地检测精度的提高，但是仍然存在极少量的漏检和误检问题；虽然基于 FCN 的企鹅数量检测方法实现了高效检测出企鹅个体并进行企鹅数量统计，但是该方法并没有达到很高的检测精准度。由于企鹅栖息地与数量检测系统内应用的算法模型是比较成熟的模型，今后可以探索更新的技术，结合实际情况应用到本系统中，进一步提升对企鹅栖息地与企鹅数量的检测精度和检测效率。

（2）由于各企鹅岛屿特征不一样，拍摄的像素不一样等情况，导致对阿德雷岛进行训练得到的模型并不能很好适用于其他岛屿的情况，因此在未来拍摄像素更加精细以及对不同岛屿进行特征规划和分类后，探索实现功能更加强大的企鹅栖息地与数量检测系统。

（3）由于目前岛屿图像数据集有限，系统的检测对比模块虽然经过设计和实现，但未能利用真实数据进行测试，所以未做结果展示，未来获取真实数据后将对该模块功能进行测试和完善。

参考文献

- [1] 徐徐.南极海冰急剧减少[N].中国科学报,2019-07-05(002).
- [2] Hickcox R P , Jara M , Deacon L A K , et al. Global terrestrial distribution of penguins (Spheniscidae) and their conservation by protected areas[J]. Biodiversity & Conservation, 2019(313).
- [3] 孙维萍,蔡明红,王海燕,邢闯,卢冰,H.U.Peter,A.Froehlich.阿德雷岛企鹅种群分布、繁殖行为及其环境影响因子分析[J].极地研究,2010,22(01):33-41.
- [4] Witharana C , Lynch H J . An Object-Based Image Analysis Approach for Detecting Penguin Guano in very High Spatial Resolution Satellite Images[J]. Remote Sensing, 2016, 8(5):375.
- [5] Larue M A , Lynch H J , Lyver P O B , et al. A method for estimating colony sizes of Adélie penguins using remote sensing imagery[J]. Polar Biology, 2014, 37(4):507-517.
- [6] Fretwell P T , Trathan P N . Penguins from space: faecal stains reveal the location of emperor penguin colonies[J]. Global Ecology & Biogeography, 2009, 18(5):543-552.
- [7] Spatially integrated assessment reveals widespread changes in penguin populations on the Antarctic Peninsula[J]. Ecology, 2012, 93(6):1367-1377.
- [8] Larue M A , Kooyman G , Lynch H J , et al. Emigration in emperor penguins: implications for interpretation of long-term studies[J]. Ecography, 2015, 38(2):114-120.
- [9] 熊登亮,陈舫益.采用无人机影像生成高原山区高精度DEM的一种方法[J].测绘与空间地理信息, 2014, 000(001):127-128,134.
- [10] Joachim Hoehle. Photogrammetric Measurements in Oblique Aerial Images[J]. photogrammetrie fernerkundung geoinformation, 2008(1):7-14.
- [11] 贺红,李香兰,程晓,朱仁斌,唐剑武,惠凤鸣,吴文会,赵天成,刘岩,康靖.基于航拍数据的南极维多利亚地企鹅源温室气体排放量估算[J].科学通报,2016,61(30):3268-3277.
- [12] He H , Cheng X , Li X , et al. Aerial photography based census of Adélie Penguin and its application in CH₄ and N₂O budget estimation in Victoria Land, Antarctic[J]. entific Reports, 2017, 7(1):12942
- [13] 冀明,张宝钢,张媛媛,李香兰,程晓.南极企鹅数量识别及变化趋势分析——基于无人机航拍的高分辨率影像[J].北京师范大学学报(自然科学版),2019,55(01):25-35.
- [14] Goebel M E , Perryman W L , Hinke J T , et al. A small unmanned aerial system for estimating abundance and size of Antarctic predators[J]. Polar Biology, 2015, 38(5):619-630.
- [15] 丁承君,闫彬,张鹏.基于边缘信息和混合高斯融合的运动目标检测[J].电视技术,2018,42(03):93-96+102.
- [16] Kim J , Lee J K , Lee K M . Accurate Image Super-Resolution Using Very Deep Convolutional Networks[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2016:1646-1654
- [17] 李昊东.基于全卷积网络改进的图像语义分割应用研究[D].南京邮电大学,2019.
- [18] Redmon J , Divvala S , Girshick R , et al. You Only Look Once: Unified, Real-Time Object Detection[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2016:779-788
- [19] 王林,张鹤鹤.Faster R-CNN 模型在车辆检测中的应用[J].计算机应用, 2018,

038(003):666-670.

[20] Ren S , He K , Girshick R , et al. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(6):1137-1149.

[21] He K , Zhang X , Ren S , et al. Spatial Pyramid Pooling in Deep Convolutional Networks for Visual Recognition[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 37(9):1904-16.

[22] Girshick R . Fast R-CNN[C]// 2015 IEEE International Conference on Computer Vision (ICCV). IEEE, 2016.

[23] 戴同武.基于深度学习的快销品图像识别系统设计与开发[D].湖南大学,2018.

[24] Yang X , Sun H , Sun X , et al. Position Detection and Direction Prediction for Arbitrary-Oriented Ships via Multitask Rotation Region Convolutional Neural Network[J]. IEEE Access, 2018, 6:50839-50849.

[25] Cheng B , Wei Y , Shi H , et al. Revisiting RCNN: On Awakening the Classification Power of Faster RCNN[J]. 2018.

[26] Long J , Shelhamer E , Darrell T . Fully convolutional networks for semantic segmentation[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2014, 37(4):640-651.

[27] 雷李义.基于深度学习的水面漂浮物目标检测及分析[D].广西大学,2019.

[28] 王森,杨克俭.基于双线性插值的图像缩放算法的研究与实现[J].自动化技术与应用, 2008, 27(7):44-45.

[29] 白胜.基于深度学习的路面修补检测的设计与实现[D].北京邮电大学,2019.

[30] 文青.基于深度学习的建筑表面裂缝检测技术研究 with 实现[D].北京邮电大学,2019.

作者简历及攻读硕士学位期间取得的研究成果

姓名：张宇轩

教育经历：

2018.9-至今 于北京交通大学软件学院软件工程专业攻读硕士学位

2014.9-2018.6 于河北师范大学软件学院软件工程专业攻读学士学位

工作经历：

2019.9-2019.11 北京滴滴出行科技有限公司 软件开发工程师

2018.8-2019.8 北京早知道科技有限公司 软件开发工程师

独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作和取得的研究成果，除了文中特别加以标注和致谢之处外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得北京交通大学或其他教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文作者签名：

张宇轩

签字日期：2020 年 5 月 25 日

学位论文数据集

表 1.1: 数据集页

关键词*	密级*	中图分类号	UDC	论文资助
Faster RCNN ; FCN; 目标检测; 语义分割; 企鹅 栖息地	公开			
学位授予单位名称*		学位授予单位代码*	学位类别*	学位级别*
北京交通大学		10004	工程	硕士
论文题名*		并列题名		论文语种*
企鹅栖息地和数量检测方法的设计与系统实现				中文
作者姓名*	张宇轩		学号*	18140691
培养单位名称*		培养单位代码*	培养单位地址	邮编
北京交通大学		10004	北京市海淀区西直 门外上园村 3 号	100044
工程领域*		研究方向*	学制*	学位授予年*
软件工程		软件工程	两年	2020
论文提交日期*	2020 年 6 月			
导师姓名*	邢薇薇		职称*	教授
评阅人	答辩委员会主席*		答辩委员会成员	
	何勇强			
电子版论文提交格式 文本 (<input checked="" type="checkbox"/>) 图像 (<input type="checkbox"/>) 视频 (<input type="checkbox"/>) 音频 (<input type="checkbox"/>) 多媒体 (<input type="checkbox"/>) 其他 (<input type="checkbox"/>) 推荐格式: application/msword; application/pdf				
电子版论文出版 (发布) 者		电子版论文出版 (发布) 地		权限声明
论文总页数*	52			
共 33 项, 其中带*为必填数据, 为 21 项。				