学 号 2017061803

密 级

哈尔滨工程大学学士学位论文

基于深度学习的图像隐写分析

院（系）名 称：计算机科学与技术学院

专 业 名 称：计算机科学与技术

学 生 姓 名：邓睿智

指 导 教 师：吴艳霞 副教授

哈尔滨工程大学

**2021**年**6**月

**基于深度学习的图像隐写分析 邓睿智 哈尔滨工程大学**

学 号 2017061803

密 级

基于深度学习的图像隐写分析

Image Steganalysis Based on Deep Learning

**学生姓名：**邓睿智

**所在学院：**计算机科学与技术学院

**所在专业：**计算机科学与技术

**指导教师：**吴艳霞

**职称：**副教授

**所在单位：**哈尔滨工程大学

**论文提交日期：**2021年6月

**论文答辩日期：**2021年6月

**学位授予单位：**哈尔滨工程大学

# 

# 摘 要

随着互联网的广泛普及以及数字化技术的快速发展，大量信息通过数字图像、音视频等载体在互联网中广泛传递，人们在享受科技带来便捷的同时却也受到信息泄密的威胁。一些不法分子为了秘密传递消息而不被发现，就使用隐写术将秘密信息嵌入到常见的载体中，这些包含隐秘信息的含密载体和不包含隐秘信息的原始载体在视觉上没有明显差异，人眼无法识别，需要使用一些特定的方法进行隐写分析，才能给出判断。在众多信息载体中，数字图像相较于其他信息载体使用领域更加广泛。故以数字图像为载体的隐写分析是目前信息安全领域一个重要研究方向。

本文通过搭建SRNet（Steganalysis Residual Network）神经网络模型对数字图像进行隐写分析，判断图片当中是否存在隐秘信息，并在原始SRNet网络模型的基础上添加了CBAM（Convolutional Block Attention Module）注意力模块，使模型可以更好的进行特征提取，从而提高模型的准确率。若隐写分析模型认为该图片中包含隐秘信息，本文将进一步利用自编码器以及生成对抗网络进行隐写去除，即破坏含密图像中嵌入的信息。

实验结果证明对于WOW（Wavelet Obtained Weights）、S-UNIWARD（Spatial-Universal Wavelet Relative Distortion）、HUGO（Highly Undetectable Stego）三种自适应空域隐写算法，本文使用的隐写分析模型能够较好的检测出图片中是否存在隐秘信息；同时在原始隐写分析模型基础上添加注意力机制之后，模型的准确率有了进一步的提升；在隐写去除部分，模型去除含密图像中隐秘信息后得到的生成图像有较好的视觉效果，和原始含密图像在肉眼上看不出区别。

关键词：数字图像；深度学习；隐写分析；隐写去除

# **ABSTRACT**

With the popularity of the Internet and the development of digital technology, a large amount of information is widely transmitted on the Internet through digital images, audio and video and other carriers. While people enjoy the convenience brought by science and technology, they are also threatened by information leakage.Some criminals to secret messages without being found, use the steganographic embedding secret information into a usual carrier, of these contain hidden information carrier and does not contain hidden information there was no significant difference between the original carrier on the vision, the human eye can't identify, need to use some specific methods for steganographic analysis, to give judgment.Among many information carriers, digital image is more widely used than other information carriers.Therefore, steganographic analysis based on digital image is an important research direction in the field of information security.

In this thesis, the SRNet (Steganalysis Residual Network) neural network model is built for image steganalysis to determine whether there is hidden information in the picture. In addition, CBAM (Convolutional Block Attention Module) attention module is added on the basis of the original model, so that the model can better extract features and improve the accuracy of the model.If the steganographic analysis model thinks that the image contains secret information, this paper will further use the AutoEncoder and GAN to carry out steganographic removal, that is, to destroy the embedded information in the secret image.

The experimental results show that for the three adaptive spatial steganography algorithms of WOW (Wavelet Obtained Weights), S-UNIWARD (Spatial-Universal Wavelet Relative Distortion) and HUGO (Highly Undetectable Stego), the steganalysis model used in this thesis can better detect whether there is hidden information in the image.At the same time, after adding attention module to the original steganalysis model, the accuracy of the model is further improved.In the part of steganography removal, the generated image obtained after the model destroys the secret information in the encrypted image has a good visual effect, and it cannot be distinguished from the original encrypted image by eyes.

**Keywords**: Digital image; Deep learning; Steganalysis; Steganographic destruction

目 录

[第1章 绪论 1](#_Toc21728)

[1.1 课题研究背景及意义 1](#_Toc21768)

[1.2 国内外研究现状 2](#_Toc8883)

[1.2.1 基于人工特征的传统隐写分析 2](#_Toc17018)

[1.2.2 基于深度学习的隐写分析 3](#_Toc12095)

[1.3 主要研究内容 4](#_Toc4431)

[1.4 论文组织结构 5](#_Toc3062)

[第2章 相关技术介绍 6](#_Toc11180)

[2.1 卷积神经网络基础理论 6](#_Toc1784)

[2.2 残差网络 8](#_Toc18811)

[2.3 生成对抗网络 10](#_Toc14762)

[2.3.1 GAN网络基本思想 10](#_Toc25385)

[2.3.2 GAN网络缺点和变种 11](#_Toc21750)

[2.4 数据集介绍 11](#_Toc24415)

[2.4.1 原始图像 11](#_Toc24743)

[2.4.2 含密图像 12](#_Toc31608)

[2.5 本章小结 14](#_Toc3705)

[第3章 基于深度学习的图像隐写分析方法 15](#_Toc11463)

[3.1 基于SRNet网络模型的图像隐写分析 15](#_Toc1751)

[3.1.1 SRNet网络结构 15](#_Toc26857)

[3.1.2 模型训练 17](#_Toc1118)

[3.1.2 模型测试 22](#_Toc30127)

[3.2 添加CBAM注意力机制后的网络模型 23](#_Toc31899)

[3.2.1 CBAM注意力机制 23](#_Toc9582)

[3.2.2 模型训练 25](#_Toc18337)

[3.2.3 模型测试与结果对比分析 26](#_Toc19123)

[3.3 本章小结 27](#_Toc1386)

[第4章 基于生成对抗网络的图像隐写去除 28](#_Toc18092)

[4.1 DDSP网络结构 28](#_Toc14112)

[4.1.1 生成器网络结构 29](#_Toc28615)

[4.1.2 鉴别器网络结构 32](#_Toc29628)

[4.2 模型训练 34](#_Toc14930)

[4.2.1 训练参数设置 34](#_Toc6818)

[4.2.2 训练输出 35](#_Toc6952)

[4.3 模型测试与结果分析 36](#_Toc15398)

[4.4 本章小结 39](#_Toc7338)

[第5章 可视化演示系统设计 40](#_Toc28439)

[5.1 登录注册界面 40](#_Toc8339)

[5.2 隐写嵌入和隐写分析界面 41](#_Toc17350)

[5.3 本章小结 43](#_Toc23918)

[结 论 44](#_Toc18466)

[参考文献 46](#_Toc23732)

[攻读学士学位期间发表的论文和取得的科研成果 49](#_Toc28129)

[致 谢 50](#_Toc9948)

# 第1章 绪论

## 1.1 课题研究背景及意义

互联网的快速发展打破了人们彼此之间的距离限制，人们可以通过互联网进行信息传递，这给人们的日常生活带来了很大便利，但互联网造福人类的同时，也带来了巨大的信息安全隐患，那就是秘密信息在传播途中容易泄密，为此人们采用各种加密方式对信息内容进行加密，这样信息载体即使被不法分子截获，由于对方不知道正确的解密方式，也就无法获取信息内容。图像隐写术是常见的加密方式之一，图像隐写术利用图像中的人眼不敏感的冗余信息，将待嵌入的秘密信息嵌入到图像中信息冗余的部分，使得嵌入的秘密信息不易引起察觉，嵌入前的原始图像和嵌入后的含密图像，两者从人眼上来看没有明显的视觉差异。

图像隐写术对信息安全传递有很大的帮助，能够很好的防止秘密信息的泄露，但它同样也是一把双刃剑，一些不法分子为了秘密传递信息而不被发现，可以使用隐写术将秘密信息嵌入到数字图像当中[1]，由于隐写前后图像的视觉不变性，不法分子就可以堂而皇之的逃过审查，从而造成秘密信息的泄露，此时作为图像隐写术的对抗技术——图像隐写分析技术就显的十分重要。

隐写分析本质上是一个二分类问题，即判断信息载体中是否存在隐秘信息，若有则结果为1，否则结果为0。由于图像隐写术在进行隐写的时候，会改变图像的某些统计特性，所以隐写分析可以利用这些统计特性进行分析并给出判断。在隐写分析领域，从隐写术方面大体上可以分成两类：针对某一类隐写算法的专一隐写分析方法、利用众多隐写术的共性进行隐写分析的通用隐写分析方法。从隐写分析使用的方法来分也可以分成两大类：传统图像隐写分析方式和基于深度学习的图像隐写分析。目前基于人工特征的传统隐写分析发展较为缓慢，在提出新隐写分析算法方面没有大的进展，人们转而将更多的注意力放在基于深度学习的隐写分析方法上，在过去的几年中人们提出的隐写分析算法绝大部分都是基于深度学习的。

由此，本文亦采用深度学习的方式对图像进行隐写分析，并且在原始网络模型的基础上，增加注意力模块，让模型能够更好的关注于对最终判断有帮助的特征，从而提高隐写分析模型的准确性。

虽然将深度学习应用于图像隐写分析领域使得图像隐写分析取得较大的成果，但是面对同样在不断进步的图像隐写术，图像隐写分析依旧远远落后。每当隐写算法和嵌入率发生改变时，模型就不能很好的判断图片中是否含有隐秘信息。面对这一局面，图像隐写去除受到了大家的关注。既然有时候不能通过隐写分析判断图像中是否含有隐秘信息，那么不妨对有含密嫌疑的图片统统进行隐写去除操作即破坏含密图像中的隐秘信息，这样即使对方接收到含密图像，由于其中的秘密信息都已经被破坏，对方也就无法知晓图片中原本嵌入的信息。只不过去隐写后的图片图像和含密图像在视觉质量方面没有太大变化，需要在实现隐写去除的同时，保证生成图像的视觉质量。

## 1.2 国内外研究现状

### 1.2.1 基于人工特征的传统隐写分析

将深度学习利用到隐写分析领域之前，隐写分析领域就已经有一些较好的传统隐写分析方法，它们一般都包含特征提取和训练分类器两部分[2]。其中特征提取依靠人工设计，早期隐写分析特征通常是特征函数质心、统计矩等低阶统计量，这些特征的维度一般不超过100维。其中较为经典的特征有2003年Harmsen等人提出基于直方图特征函数质心[3]（Histogram characteristic function center of mass, HCFCOM）特征，Lyu等人于同年提出基于小波分解的高阶统计量特征[4]，Chen等人于2006年提出的基于经验矩阵的特征[5]等等。

随着隐写术的不断发展，如WOW[6]算法、SUNIWARD[7]算法、HUGO[8]算法、MiPOD[9]算法等自适应隐写算法的出现，隐写图像保存的图像高阶统计特性更加复杂，隐写分析所需要的特征也随之复杂化，由此，基于对图像邻域复杂相关性进行建模处理的高阶统计量特征成为主流，此类算法共分为两步：首先对图像进行不同的滤波操作得到多种残差图像；接下来分别从各个残差图像中计算描述近邻像素间相关性的共生矩阵[10]、投影直方图[11]、LBP[12]（Local Binary Pattern）等作为特征，滤波操作目的是去除载体图像自身内容，避免载体图像内容对隐写分析造成干扰，同时保留隐写引入的高频噪声，通过组合各种不同的残差特征，可以使得算法提取更加丰富的隐写细节，使得检测性能有较大提升，此类特征在维度上一般达到了万维级别。此后，为了提升隐写分析模型对自适应隐写术检测的准确率，人们在提取特征的过程中加入选择通道。Tang等人首先提出了利用选择信道信息的自适应隐写分析[13]，该隐写分析算法在WOW隐写术上有较好的检测效果。2014年Denemark等人在Tang的基础上提出了一种新的通用自适应隐写分析特征[14]，该方法对比传统隐写分析方法，在HUGO、WOW等隐写算法中检测的准确率有进一步提升。

能否较好的获取特征直接关系到结果的准确性，因此在传统隐写分析领域，绝大部分的注意力都集中在如何设计和提取特征，然而特征设计和提取对研究者有较高的要求，它要求研究者在隐写分析领域有深厚的知识以及丰富的经验；此外由于特征提取和分类器优化这两个部分是相互独立的，因此无法同时进行优化，这对隐写分析的效率也有一定影响。

### 1.2.2 基于深度学习的隐写分析

基于深度学习的图像隐写分析出现较晚，相较于传统隐写分析，深度学习并不分段进行，而是采用端到端的学习方式，将特征提取和分类器训练放在一起同步训练，并且将特征提取和分类任务交给模型通过训练进行自主学习。

最早将深度学习应用到隐写分析领域的是Tan等人于2014年提出的TanNet[15]，这个网络结构比较简单，只包含三层卷积和一层全连接。在对第一个卷积层卷积核参数初始化方面TanNet也有三种不同的处理方式[16]：第一种方式是随机初始化卷积核；第二种方式是通过用KV高通滤波核乘以随机初始化的卷积核的方式来初始化卷积核参数；第三种方式是使用KV高通滤波核初始化第一层卷积核以及使用AutoEncoder预先训练每一个卷积层。训练结果显示第三种卷积核初始化方式模型得到的结果最优，第二种次之，第一种随机初始化卷积核模型无法成功收敛。训练完毕后作者将该模型与传统隐写分析中的SPAM和空域富模型[17]（Spatial Rich Model，SRM）方法进行对比，实验结果显示TanNet的性能强于SPAM但是远弱于SRM。虽然TanNet的性能并不优秀，但却显示了深度学习在隐写分析领域巨大的潜力，是将深度学习应用到隐写分析领域的开山之作。

2015年，Qian等人提出了QianNet[18]，该网络最开始是一个预处理层，预处理层中通过使用固定的KV高通滤波核过滤掉原始图像的内容，减少图像本身内容对隐写分析的影响，同时保留高频噪声部分，从而提高隐写信号和宿主图片的信噪比。预处理层之后是五层卷积层，作者观察隐写噪声的特点发现在卷积层中使用Gaussian激活函数对最终的实验结果有较大的帮助。此外考虑到高频噪声的丢失问题，作者在池化阶段均采用平均池化而没有采用最大池化。QianNet网络的准确率高于TanNet，和SRM算法相比相差无几。

2016年，Xu等人提出了XuNet[19]，XuNet网络利用网络前端的绝对值函数、Tanh激活函数、1 × 1卷积以及批量标准化等操作使特征图变得紧凑化。此外XuNet在网络最开始的预处理层中同样采用固定的KV高通滤波核对图像内容进行压缩，提高隐写信号和宿主图片的信噪比，XuNet相较于之前网络的准确率有较大的提升，在BOSSbase1.01数据集[20]上的准确率超过了传统隐写分析方法SRM。

2017年提出的YeNet[21]被认为是深度学习在隐写分析领域中一个突破性的进展，因为其提出的检测器显著的改进了之前已有的空域隐写分析检测器。YeNet有三个特点：(1) 提出了一种新的激活函数TLU (Thresholded Linear Unit)，TLU能够提高模型的收敛速度也能更好的适应隐写噪声分布的特征规律。(2) 在模型的第一层采用30个5 × 5大小的SRM空域富模型卷积核初始化参数。(3) 引入选择通道，将选择通道和卷积神经网络相结合，进一步提升网络的性能。此外YeNet也指出大型数据集对于训练深度较深的神经网络的重要性，以及使用自适应优化器特别是AdaDelta梯度下降变体的优点。YeNet的准确率远远超越了SRM和maxSRMd2等传统隐写分析方法。

2019年M.Boroumand等人提出的SRNet[22]网络模型是目前隐写分析性能较好的隐写分析模型。SRNet在YeNet的基础上做了进一步优化，网络前端预处理层的参数不再人为固定，而是让网络通过训练实现自主学习。特别需要注意的是，网络前端能否很好的进行特征提取将直接关系到模型的检测性能，所以网络前七层取消了池化操作（池化操作本质上是一种低通滤波，这会损失大量隐写高频噪声信号，对隐写分析不利）。SRNet不仅在空域隐写术中表现出优秀的性能，在JPEG域同样有很好的准确率。本文中图像隐写分析部分就是复现SRNet网络模型，并且在原始SRNet网络模型的基础上添加注意力机制进一步提高模型性能。

## 1.3 主要研究内容

本文的主要研究内容分为两部分：分别是基于深度学习的图像隐写分析和基于自编码器以及生成对抗网络的图像隐写去除。

1. 基于深度学习的图像隐写分析。首先准备模型需要的原始数据集，将数据集中图片全部缩放为指定尺寸，按照一定的比例划分成训练集、验证集和测试集，并且选择若干隐写术按照一定嵌入比特率向原始图片中嵌入信息从而生成含密图像。之后编写代码搭建SRNet网络模型，训练网络模型。原始SRNet网络模型训练完成之后，修改网络结构部分的代码，在原始SRNet网络的基础上添加CBAM注意力模块[23]，再次训练网络模型，观察改进网络结构前后，模型准确率的变化。
2. 基于自编器和生成对抗网络[24]（Generative Adversarial Nets，GAN）的图像隐写去除。由于目前隐写分析算法存在较大的局限性，因此不妨转换思路，不再执着于判断图像中是否包含隐秘信息，而是对有含密嫌疑的图片进行隐写去除，前提是保证隐写去除后的图片，其视觉质量和原始图片没有太大变化。

## 1.4 论文组织结构

本文一共分为5章来介绍相关工作。

第1章是绪论。主要介绍了本文的研究背景以及意义；之后介绍了国内外目前在深度学习图像隐写分析领域的研究现状；最后介绍了本文的研究内容以及论文的组织结构。

第2章是相关技术的介绍。主要介绍本文所使用的一些相关技术。第一节介绍卷积神经网络中诸如卷积、池化这样的基础知识；第二节介绍隐写分析网络模型中需要用到的残差网络结构；第三节介绍隐写去除中用到的GAN网络基本概念；第四节介绍了模型训练所采用的数据集以及隐写术。

第3章是基于深度学习的图像隐写分析方法实现。首先介绍了原始的SRNet网络模型结构，模型训练参数和训练输出，之后介绍CBAM注意力机制以及将注意力模块添加到原始SRNet网络模型中从而实现模型改进。最后通过对比实验，验证添加注意力模块后的模型相对原始SRNet模型有更高的准确率。

第4章是基于GAN网络的含密图像隐写去除。主要介绍DDSP[25]隐写去除模型中Generator和Discriminator的结构，之后介绍了模型训练输出和测试结果。

第5章是演示系统设计，简单介绍演示系统GUI界面的设计过程。

# 第2章 相关技术介绍

## 2.1 卷积神经网络基础理论

卷积神经网络模型一般有5个组成部分，分别是：卷积层、池化层、激活函数层、批量标准化层和全连接层，下面将详细阐述它们的原理。

**卷积层**：在卷积神经网络中卷积层是最重要的一部分，其工作原理是利用若干个卷积核（卷积核的大小通常是3×3，5×5）对上层输入的数据进行卷积计算从而生成特征图，生成的特征图作为后续层的输入，其计算过程如公式（2-1）所示。

 （2-1）

在公式（2-1）中，表示二维输入数据在点的图像像素值，表示卷积核尺寸为的卷积核在点()的权重值，而表示该层的偏置大小，表示网络所使用的激活函数，表示点()经过卷积计算后所得到的输出大小，和表示输入图像分辨率的大小。其计算过程如图2.1所示。

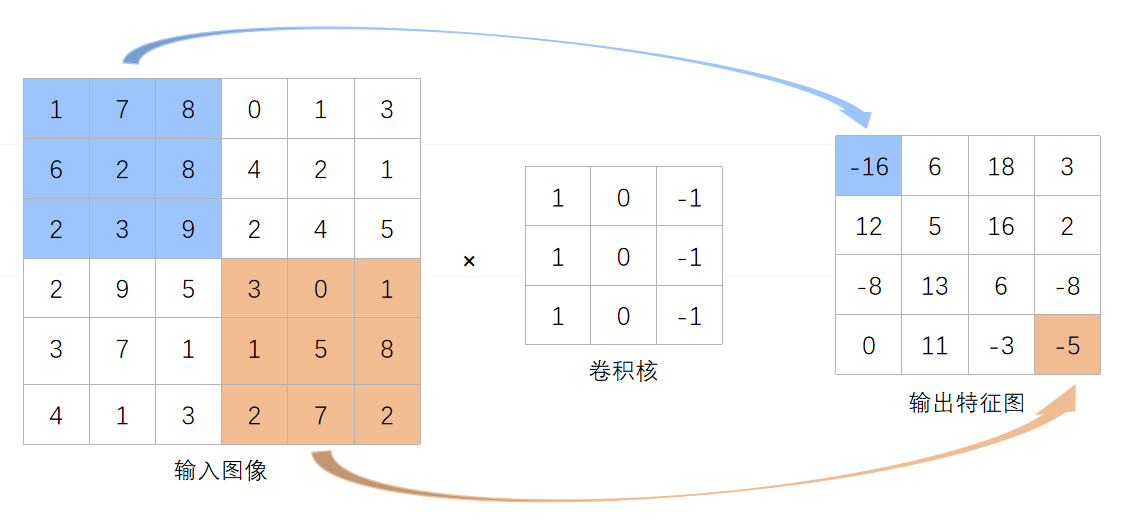


图2.1 卷积操作示意图

**池化层**：池化层的目的是降低特征图的大小和维度，使卷积层提取到的特征更加紧凑化，这样做可以大大减少数据的计算量，同时提高模型在位移和形变方面的鲁棒性，其公式如（2-2）所示。

 （2-2）

在公式（2-2）中表示池化函数，表示池化输入，表示池化输出。一般在卷积神经网络中有两种池化函数：最大池化和平均池化。最大池化是选择局部区域的最大值作为输出，而平均池化是计算局部区域的平均值作为输出，池化的主要作用是将输入特征映射聚合成一个紧凑的表示。此外，通过将特征集中到一个较小的尺寸中可以获得图像中的大距离相关性。最大池化和平均池化操作示意图如图2.2所示。

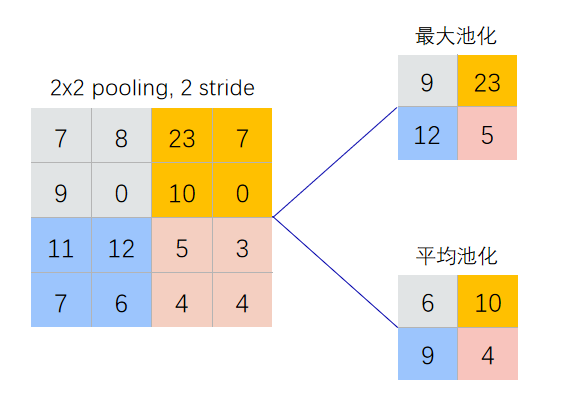


图2.2 最大池化和平均池化示意图

**激活函数层**：激活函数层的作用是对输入特征图进行非线性映射，它对于卷积神经网络而言是很重要的，因为如果没有非线性映射，任何层数的神经网络都等于只有一层的网络。此外，非线性映射的存在使得卷积神经网络能在自然图像中提取更加复杂的相关性。常用的激活函数有Sigmoid、ReLU、Tanh等，其函数图像如图2.3所示。

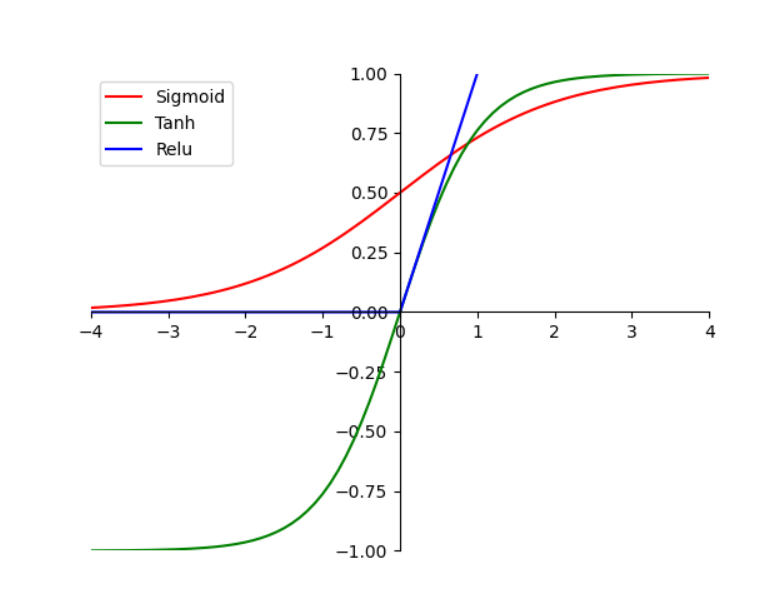


图2.3 常见激活函数图像

**批标准化层**：批标准化的作用是将一批数据的每一个数据项归一化为，归一化公式如（2-3）所示。

 （2-3）

β和γ是批标准化的参数，其中的可由进行表示，具体过程见公式（2-4）。

 （2-4）

在公式（2-4）中和分别表示该批数据的均值和方差，是一个极小值，其作用是为了防止分母为0，批量标准化可以使数据远离饱和区，这样可以使批处理归一化后的神经网络对参数初始化相对不敏感，其收敛速度相对于没有批处理标准化的网络更快。

**全连接层**：全连接层一般出现在卷积神经网络的后端，起到一个分类器的作用，全连接操作本质上是一个矩阵乘法，也可以看成是采用全部连接方式的卷积层，其卷积核的尺寸为1×1。卷积神经网络中虽然全连接层只在末端出现且层数远少于卷积层的数量，但网络中大部分参数还是来自于全连接层，所以模型整体呈现出头轻脚重的形式。

## 2.2 残差网络

对于卷积神经网络而言，当网络层数较少时适当增加网络的深度，网络模型的性能会相应提高[26]，但这并不意味着在浅层网络的基础上不断堆叠卷积层，网络深度变的越深，其性能就越好。相反，若神经网络深度过深的话，就会出现三个严重的问题：过拟合、梯度弥散和梯度爆炸[27]、退化。

过拟合是指网络在训练集上有很好的准确率，但是在测试集上表现很差。网络模型需要处理的任务比较简单而网络本身却很复杂，或者训练集规模太小等因素都会引发过拟合现象。当出现过拟合现象时，一般可以采用Dropout、L1正则、L2正则、增大训练集等方式处理。

梯度弥散和梯度爆炸是在梯度反向传播时出现的。向网络中输入数据并经前向传播获得损失值之后，对损失函数求导，反向传递梯度优化网络参数，由于梯度求解的链式法则，靠近输出层的隐藏层获得的梯度较大，参数更新较快，但是对于远离输出层的隐藏层，其求解得到的梯度很小，甚至无限趋近于0，这就导致无论怎么训练，网络模型都不收敛或者收敛效果很差。梯度爆炸和梯度弥散类似，只是反过来，当浅层网络层的梯度通过训练而增大，后面深层网络层的梯度呈现指数级的增大，从而发生梯度爆炸现象。当出现梯度弥散时，可以采用如：不使用Sigmoid、Tanh激活函数，改用ReLU、MaxOut激活函数；增加批标准化（Batch Normalization）操作；当出现梯度爆炸时，可以采用梯度截断法（clip gradient），设定一个梯度阈值，规定若求解得到的梯度超过了这个阈值，则进行截断处理。

网络退化是指随着网络深度的增加，其性能反而出现退化，深度较深的网络的性能不如深度较浅的网络模型（如图2.4所示[24]），过拟合现象和梯度弥散、爆炸现象都有相应的办法处理，那网络退化该如何处理？答案就是ResNet[28]（Deep Residual Learning for Image Recognition）深度残差网络。

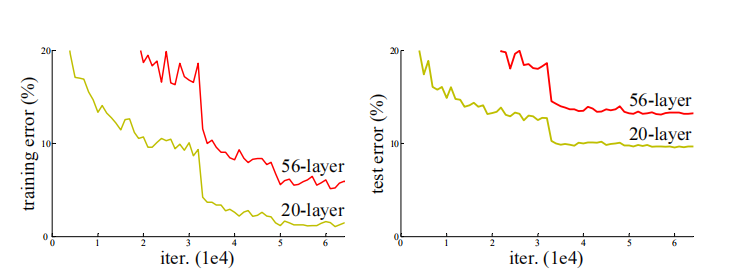


图2.4 网路性能退化示意图

由图2.4可知，不论是在训练集上（左图）还是在测试集上（右图）56层网络模型的性能反而不如20层网络的性能，出现了网路退化现象，并且要强调的是这种网络退化现象并不是由过拟合引起的。

何凯明等人于2015年提出的深度残差网络（ResNet）解决了这个问题，ResNet提出了一种叫做快捷连接（short connection）的结构，如图2.5所示，在原始串行网络的基础上添加了一个恒等映射（identity mapping）。

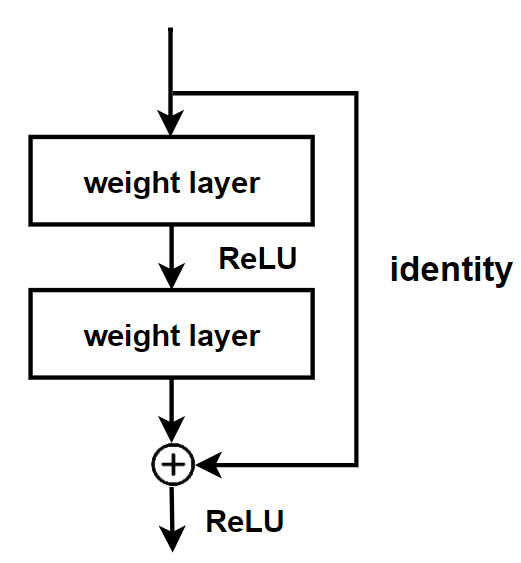


图2.5 残差网络结构示意图

网络模型中网络层可以等价为一个函数映射：，其中是该层输入，是该层的输出。而在ResNet残差模块中可以表示成：，即在输入输出之间增加了一个恒等映射，之前网络需要学习的是，而现在网络只需学习这一部分残差映射。理论证明要比更容易学习。

在ResNet网络中有两种残差结构，如图2.6所示，其中左图是针对ResNet-34所设计的，右图是针对ResNet-50/101/152等深度较深的网络设计的。ResNet有非常优越的性能，在2015年的ILSVRC竞赛中获得了第一名，其top-5错误率只有3.57%，同时参数量却比VGGNet低。ResNet打破了之前深层网络遇到的瓶颈，同时因为它非常好的推广性，很多网络模型都借鉴了ResNet的思路。

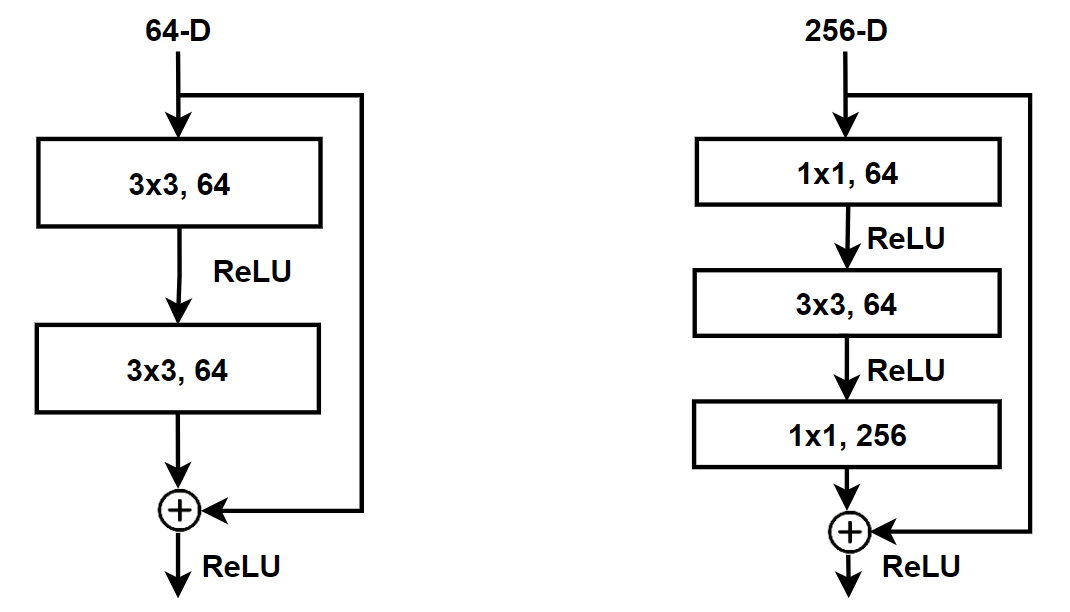


图2.6 ResNet两种残差结构示意图

## 2.3 生成对抗网络

### 2.3.1 GAN网络基本思想

GAN生成对抗网络是Goodfellow等人于2014提出的生成式网络。该网络包含两个子网络：生成器网络（Generator, G）、鉴别器网络（Discriminator, D），它们可以由自编码器、卷积神经网络等构建。

GAN网络训练本质上是一个零和博弈的过程，博弈双方的利益之和是一个常数，一方占据上风时另一方势必处于下风。其基本结构如图2.7所示，网络的输入为随机噪声记为，其首先经过生成器网络G的处理并得到输出信号，之后将与真实数据分别输入到鉴别器网络D中得到和。

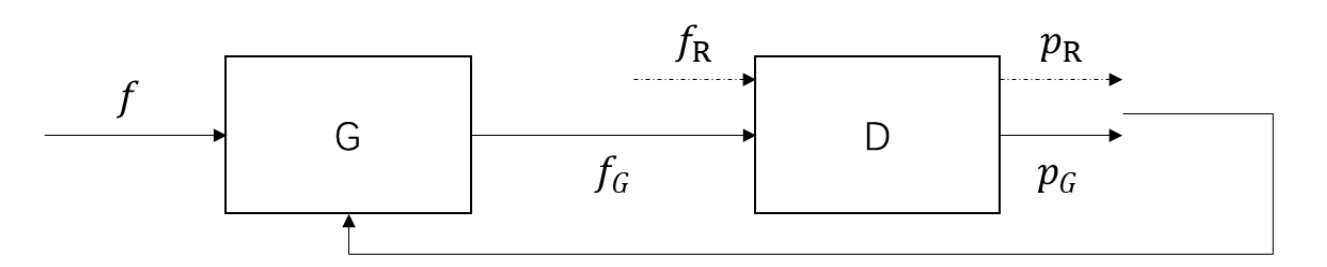


图2.7 GAN网络结构示意图

鉴别器网络D的目标很简单，就是判断输入的数据是真实数据还是Generator生成的假数据。通常鉴别器网络的输出是一个介于0和1之间的常数（由Sigmoid函数实现），该常数表示输入的数据是真实数据的概率。理想状态下，真实数据输入D网络之后得到的输出应该等于1，由G网络生成的假数据输入D网络后得到的输出应该等于0，这表明D网络可以很好的鉴别出输入的是真实数据还是虚假数据。

生成器网络G的目标也很简单，就是希望生成数据的分布可以无限接近甚至等于真实数据的分布，从而可以迷惑D网络，令其无法区分出和，在训练G网络时，需要利用D网络的输出生成损失，从而优化G网络的参数。当和大致相等时，此时认为Generator网络和Discriminator网络的性能达到了最优，网络处于纳什均衡状态，GAN网络训练结束。

在训练GAN网络时，采用Generator网络和Discriminator网络交替训练的方式。首先从一个简单的分布中随机选取一个噪声，之后开始训练G网络：将输入G网络生成，冻结D网络的参数并将生成的和真实数据输入到D网络中，设置，，计算损失，反向传导梯度对G网络参数优化，G网络训练结束。之后开始训练D网络，首先获得G网络的输出并切断G网络的梯度传导，将和真实数据输入D网络，设置，，计算损失值，调整D网络的参数。

### 2.3.2 GAN网络缺点和变种

GAN网络自提出以来就引起很多学者的研究兴趣，获得了很大的成功。再加上GAN网络无监督学习的性质，不需要额外为数据创造标签，这使得GAN网络的应用场景变得很广泛。但GAN网络同样有诸如：模式崩溃、模型难以收敛等问题。模式崩溃一般在训练不稳定的时候会发生，即使长时间训练网络，模型也很难有好的效果。由于G网络的梯度是来自于D网络的输出，若Discriminator网络的性能远远强于Generator网络的性能，那么无论G网络怎么训练，都会被D网络识破，G网络的能力就不会进步。反过来，若G网络的性能远远强于D网络，那么不论G网络输出的是什么，都可以很容易的迷惑D网络，这样G网络的性能同样不能有进一步的提升。

针对GAN网络出现的一些问题，很多学者基于原始GAN网络提出了许多不同的变种。其中f-GAN[29]提出f散度，对于f作出的不同选择对应于主流的不同散度，因此可以认为GAN是f-GAN的一种特殊情况，此算法使得GAN网络训练变得更加稳定。WGAN[30]解决了GAN训练难收敛的问题，作者从根源上分析了GAN存在的问题，通过对生成样本和真实样本加噪声使得两个分布产生重叠，提出使用Wassertein距离替代JS、KL散度，从理论上可以解决训练不稳定这一问题。SRGAN[31]使用内容损失和对抗性损失作为感知损失函数而不采用原始的损失函数，这样能够生成纹理更加丰富的图像，在图像还原和修复领域取得了优异成绩，本文中隐写去除所使用的DDSP模型就借鉴了SRGAN网络的结构。

## 2.4 数据集介绍

### 2.4.1 原始图像

模型训练、验证、测试都需要相应数据，本文使用的数据集是BOSSbase1.01数据集加上COCO数据集[32]中的10000张图片。其中BOSSbase1.01数据集包含10000张图片，是隐写分析领域中公认使用的数据集；COCO数据集中的图片同样是大小为512×512的pgm格式的灰度图，需要将图片缩放到256×256大小，缩放后的图片的格式依旧是pgm格式。总的来说，数据集中包含了20000张大小为256×256的pgm格式的灰度图，数据集中部分图像展示如图2.8所示。

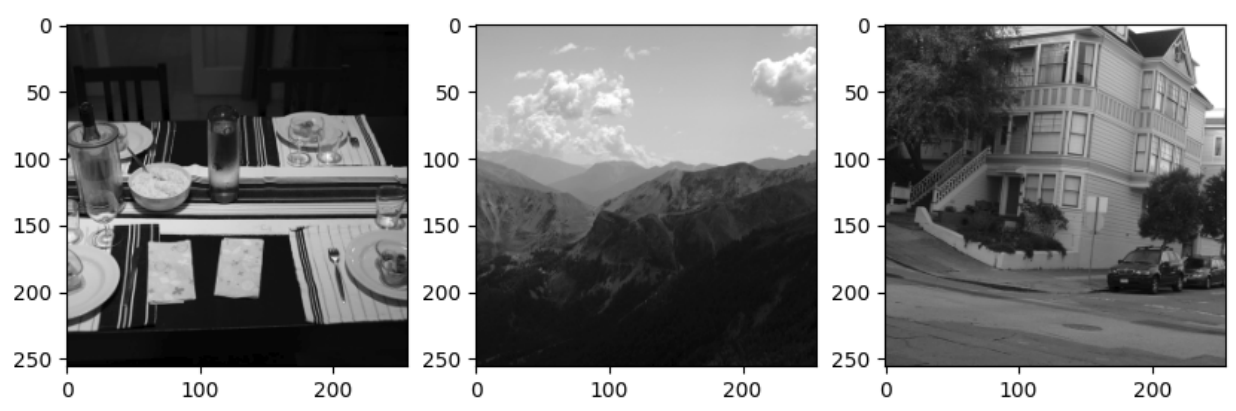




图2.8 数据集部分图片展示

### 2.4.2 含密图像

整理好原始图像之后还需要利用隐写术向原始图像中嵌入信息生成含密图片。本文使用三种隐写术：WOW、S-UNIWARD、HUGO，这三种隐写术都是图像空域隐写算法，也是为数不多的即使采用深度学习方式也很难鉴别出来的传统隐写术。在嵌入率方面，每一种隐写术都采用三种不同的嵌入率：0.4bpp、0.7bpp、1.0bpp，其中嵌入率越小，对模型而言检测难度越高，即嵌入率为0.4bpp的图片最难检测，1.0bpp的图片最容易检测。针对三种隐写算法，每种隐写算法有三种不同的嵌入比特率，所以需要生成9类不同的含密图像数据集。

故数据集共有原始图像20000张、含密图像180000张。数据集划分方面，按照14 : 1 : 5的比例对数据集进行划分，即训练集有14000张图片、验证集1000张图片，测试集5000张图片。图2.9展示了若干隐写前后的图像以及两者的残差图像。

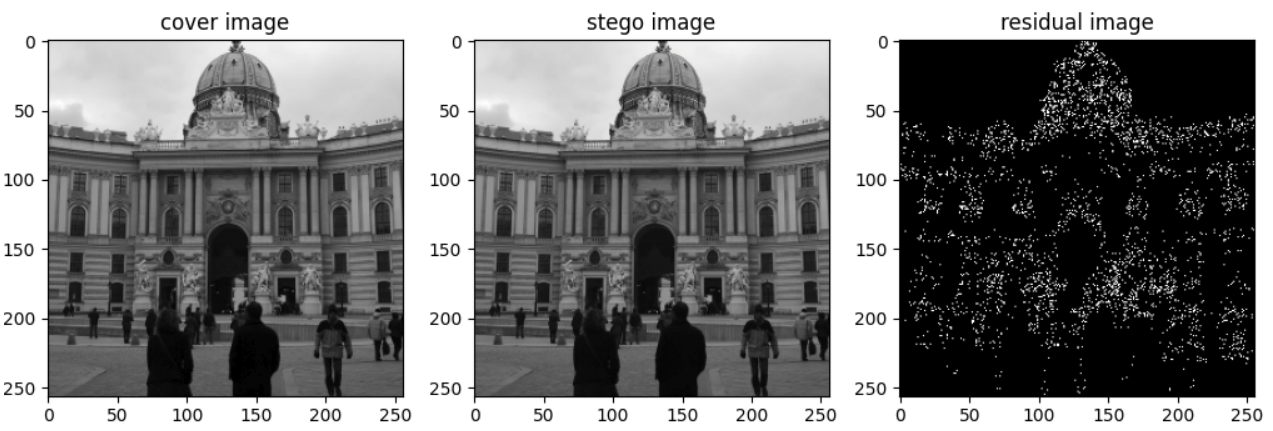
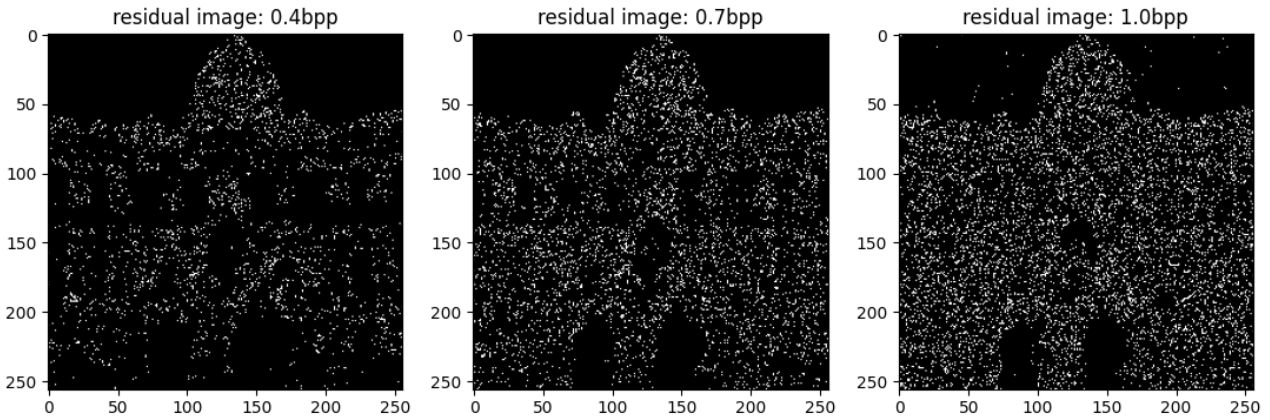




图2.9 隐写嵌入前后对比（WOW隐写术为例）

图2.9中左侧是未嵌入水印的原始图像，中间是嵌入隐秘信息后的含密图像，右侧是两者的残差图像。观察左侧和中间的图片，发现两者一模一样，肉眼看不出区别。在嵌入率方面，嵌入率越大，对原图片的修改就越大，对于不同嵌入率得到的原始图像和嵌入后图像的残差图像如图2.10所示。



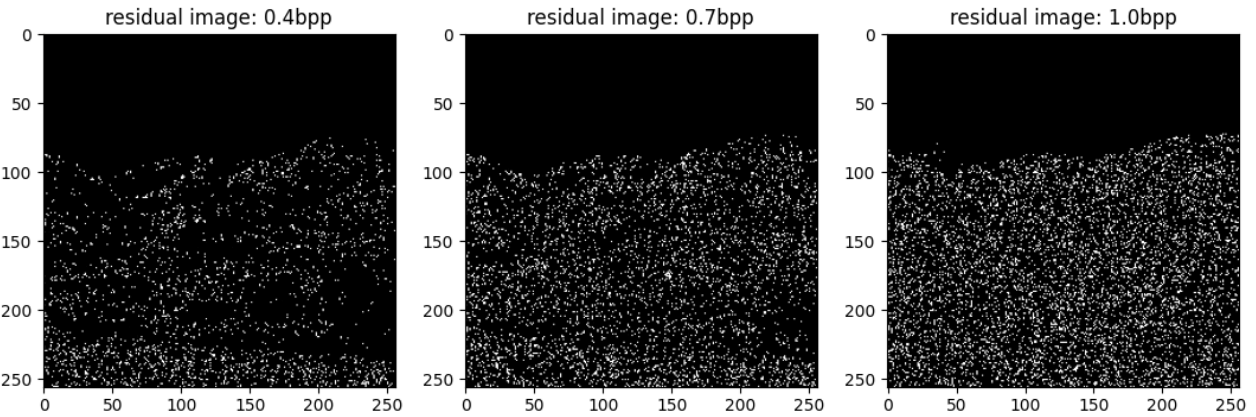


图2.10 不同嵌入率下的残差图

图2.10中从左到右图像的嵌入率分别对应为0.4bpp、0.7bpp、1.0bpp，由图片可知：嵌入率越大，残差图像越明显。

## 2.5 本章小结

本章介绍了一些本文后续会用到的技术，分别是卷积神经网络、残差网络结构、生成对抗网络以及如何生成模型训练需要的原始图像和含密图像。其中残差网络结构主要在隐写分析以及隐写去除方中使用，GAN生成对抗网络主要在隐写去除当中用于提高生成图像的视觉质量。

# 第3章 基于深度学习的图像隐写分析方法

M.Boroumand等人于2019年提出的SRNet网络模型是目前隐写分析性能较好的隐写分析模型。本文代码复现了SRNet网络结构，同时在原始SRNet网络的基础上添加了CBAM注意力模块，使得模型能更好的提取对结果有帮助的特征，提高模型的准确率。本章首先介绍SRNet网络模型以及其网络结构设计原理，之后介绍CBAM注意力机制，以及将CBAM注意力机制添加到原始SRNet网络之后新型网络模型的结构。详细说明训练过程中的训练参数，并展示最终模型测试成果。

## 3.1 基于SRNet网络模型的图像隐写分析

### 3.1.1 SRNet网络结构

利用隐写术对原始图片进行隐写嵌入，保持隐写前后图片相同的视觉效果是隐写术需要遵守的基本要求之一，这就意味着利用WOW、HUGO、S-UNIWARD隐写术进行隐写嵌入对原始图像的改动是极其微小的，即使将嵌入率提升到1bpp，嵌入前后图像的峰值信噪比依旧在50以上（通常峰值信噪比在40以上就可以认为修改后的图像和原始图像很接近）。

对于隐写分析而言，图像本身的内容信息是无用的，真正对隐写分析有帮助的是图像中在隐写嵌入时引入的高频噪声信号，所以在SRNet网络发表之前的大部分网络模型都在网络最前端加一个预处理层，该预处理层的作用就是对图像进行高通滤波，过滤掉图片中的基本图片内容，保留其中高频细节以及噪声部分，并将得到的高频噪声输入后续网络进行处理。具体的做法有：利用固定的KV高通滤波核进行滤波；使用SRM空域富模型卷积核初始化网络前端的卷积核等。但不论采用哪种方式，这里都涉及到人为知识的使用，不论是KV高通滤波核还是SRM空域富模型的卷积核都是前人研究得出的，可能适用于某些数据集和隐写术，但是对某些数据集或隐写术并不适合。一般来说最好的高频残差获取方式应该是网络模型通过训练自己学习到的，而不是人为确定，这样即使更换了数据集和隐写术。模型也可以通过自主学习获得高频噪声残差。

宏观上来看，SRNet网络模型可以分为三部分，第一部分任务是进行高频噪声的提取；第二部分的任务是特征提取，即提取对结果有帮助的特征、第三部分是一个分类器，利用上一部分提取的特征判断输入的图像是原始图像还是含密图像，若是原始图像则输出0，否则输出1，SRNet网络结构如图3.1所示。

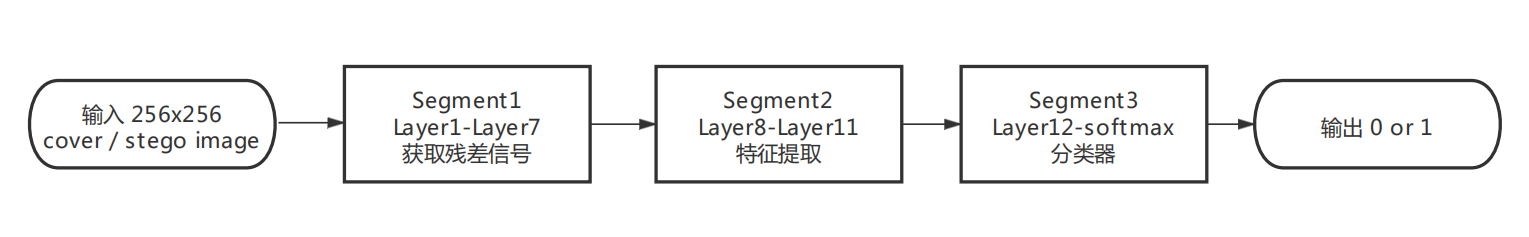


图3.1 SRNet网络简要流程图

更具体的网络结构图如图3.2[22]所示：

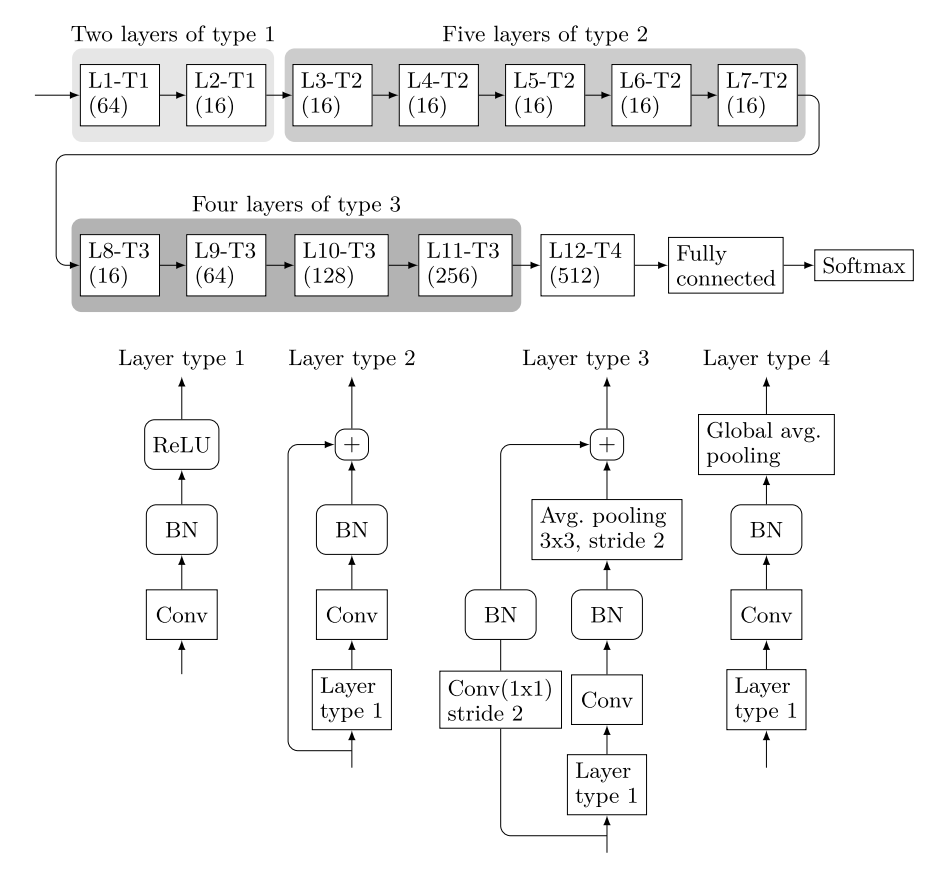


图3.2 SRNet网络结构示意图

图3.2上半部分描述了SRNet网络整个流程结构。整个网络可以划分成三段四类（即按照功能模块可以划分为3段、按照网络结构可以划分成4类），第一段对应网络中前两个阴影块也就是Layer1到Layer7；第二段对应第三个阴影块也就是Layer8到Layer11；最后一段就是剩余的所有网络层，也就是Layer12及其之后的网络层。每一个白色方块中的L表示层数、T表示结构类型、括号中的数字表示使用的卷积核的数量（例如第一个方块：L1-T1（64）表示这是第一层Layer 1，网络结构是第一种类型Type1，使用了64个卷积核）。

图3.1下半部分描述了网络中四种模块的具体结构。第一种类型网络结构Layer type 1中，输入先经过卷积、后进行批量标准化Batch Normalization，最后经过ReLU激活函数即得到输出；第二种类型网络结构Layer type 2是一个残差结构，输入首先经过一个和Layer type 1一样的网络结构，之后再经过卷积以及批量标准化，最后加上一开始的输入即得到最终的输出；第三种类型网络结构Layer type 3同样是一个残差结构，只不过并不是和Layer type2一样直接加上开始的输入，输入的tensor兵分两路，一路先经过Layer type 1网络结构，之后再经过卷积、批量标准化以及池化核大小为3×3步长为2的平均池化，另一路经过卷积核大小为1×1步长为2的卷积，再进行批量标准化，将这两路得到的结果相加即为最终的输出；第四种类型网络结构Layer type 4中，输入先经过一个Layer type 1网络结构，之后再经过卷积、批量标准化以及全局池化即得到最终的输出。第12层结束之后，网络中卷积部分就结束了，之后接上一层全连接以及softmax激活函数即可得到网络最终的输出。

SRNet网络的输入是大小为256×256的灰度图，网络中所有的卷积层使用的都是大小为3×3步长为1的卷积核，使用的激活函数都是ReLU函数。在第12层结束之后，输出的是包含512个16×16大小的特征图，将其中每一个16×16大小的特征图中的值加和求平均值，从而将之前512个16×16大小的特征图转变成一个512维的向量，并将该向量作为后续全连接层的输入。

SRNet网络中最关键的部分是前7层噪声残差提取部分。因为平均池化本质上是一种低通过滤，它强化了图像本身的内容并通过平均相邻嵌入变化来抑制类噪声隐写信号，这对传统的计算机视觉任务如模式识别、图像分类、目标检测等是有帮助的，但是对隐写分析是有害的，因为隐写分析真正需要的信号是隐写噪声，而不是图像本身内容，低通滤波会导致大量的高频信息丢失，从而去除图像中之前隐写嵌入带来的高频噪声信号。于是SRNet网络在前7层都没有使用池化操作，直到第8层才使用，这样可以减少隐写信号的丢失。SRNet中的所有滤波器都是通过端到端训练过程随机初始化和学习的，而不是人为固定的。这使得网络有更好的可扩展性，即使面对不同的隐写方法，网络依旧有较好的检测性能。

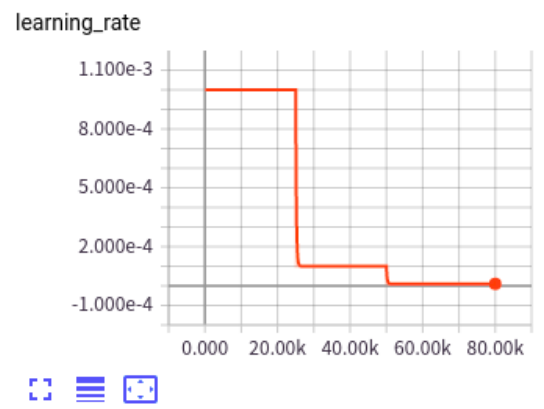
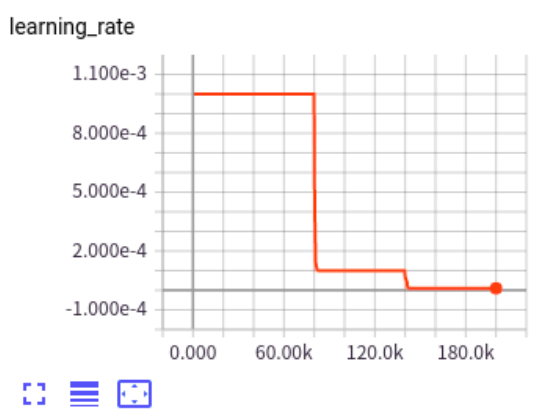
### 3.1.2 模型训练

在网络训练超参数设置方面，模型采用的优化器是Adam优化器的变种Adamax[33]（也是一种随机梯度下降优化器Stochastic Gradient Descent Optimizer）；损失函数采用交叉熵损失函数（Cross Entropy Loss，在分类问题中损失函数常用交叉熵损失函数）；训练的batch size大小为8对cover-stego图像，也就是每一次向网络中输入16张图片，其中8张原始图像8张含密图像。训练图像在输入网络之前还需要进行数据增强，本文采用90度整数倍旋转和水平镜像翻转两种方式。此外为了防止网络出现过拟合，本文采用L2正则对权重参数进行惩罚，将权重参数的L2正则作为损失函数的一部分。在网络参数初始化方面，所有卷积操作weight权重参数初始化均采用He initializer并且设置2×10-4大小的L2正则项，bias偏移量均初始化为0.2并且不设置L2正则项。对于所有的全连接层，其weight权重初始化采用均值为0，标准差为0.01的高斯分布，不设置bias偏移量。

在训练迭代iterations次数方面，不同嵌入率模型训练迭代的次数不相同。考虑到嵌入率为1.0bpp（bit per pixel）的模型更容易训练，因此本文先训练嵌入率为1.0bpp的模型，训练的iterations数量为200k（20万）次，一开始学习率设置为1e-3，迭代次数iterations到达120k时，学习率变为1e-4，当iterations为160k时，学习率变为1e-5。嵌入率为1.0bpp的模型训练完成之后，开始训练嵌入率为0.7bpp的模型，这里可以利用之前训练好的嵌入率为1.0bpp的网络模型权重文件初始化网络模型，这样可以减少模型训练的时间，故在训练嵌入率为0.7bpp和0.4bpp的模型时，iteration的数量减少为80k次，学习率的值同嵌入率为1.0bpp时一致，只是学习率变化的边界修改为20k和50k，即0k-20k时学习率的值为1e-3，20k-50k时学习率为1e-4，50k-80k时学习率为1e-5，关于学习率的变化情况详见图3.3。训练完嵌入率为0.7bpp的模型之后，同理按照和0.7bpp模型相同的方式训练0.4bpp的模型即可。

SRNet网络训练有9个不同的数据集（3种不同的隐写算法，每种隐写算法又有三种不同的嵌入率，共9个数据集），每个数据集分别训练一个模型，并且按照训练的难易程度从最简单的1.0bpp开始训练，之后是0.7bpp、0.4bpp。但本文一开始并没有按照1.0bpp-0.7bpp-0.4bpp的顺序进行训练，原因很简单，那就是需要训练9个模型，训练的时间太久。最初设想只训练嵌入率为0.4bpp的模型，若模型收敛的很好，就不用再训练嵌入率为0.7bpp和1.0bpp的模型，因为连鉴别很困难的嵌入率为0.4bpp的含密图片网络都可以很好的识别，嵌入率更大的0.7bpp和1.0bpp的含密图片模型肯定也能区分出来。但实际尝试过后发现，直接训练0.4bpp的模型，最终结果很不理想，模型在验证集上的准确率只有65%左右，大大低于预期。于是调整训练思路，从易到难，逐步递进，最终模型有比较好的准确率。

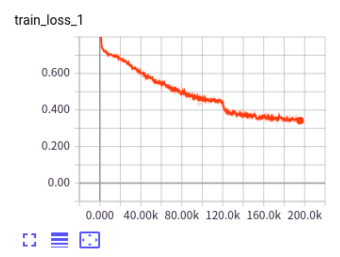
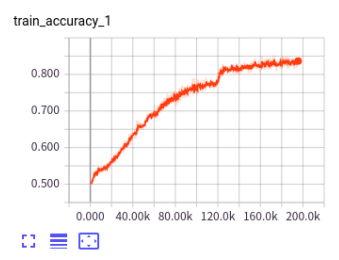
此外为了判断模型是否过拟合以及iterations为何值时模型训练得到的效果最好，规定每迭代训练5000次，就在验证集上进行测试并记录其accuracy和loss，同时保存当前网络模型参数。



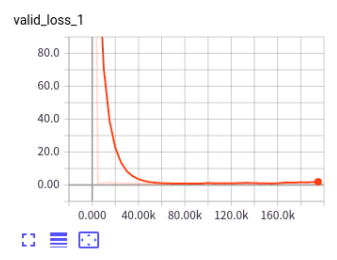
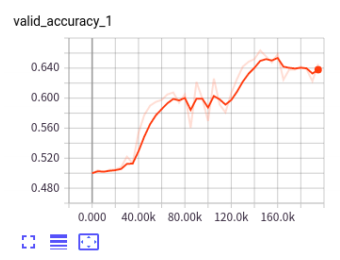
（a）训练200k学习率变化图 （b）训练80k学习率变化图

图3.3 学习率变化示意图

直接训练嵌入率为0.4bpp的模型，其accuracy和loss变化如图3.4所示，其中左上角、右上角、左下角、右下角分别为模型在训练集上accuracy变化示意图、在训练集上loss变化示意图、在验证集上accuracy变化示意图和验证集上loss变化示意图。



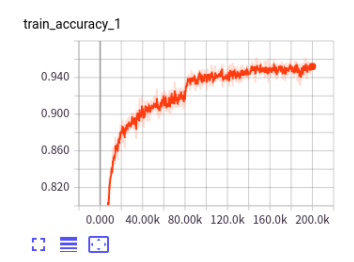
（a）训练集accuracy变化图 （b）训练集loss变化图



（c）验证集accuracy变化图 （d）验证集loss变化图

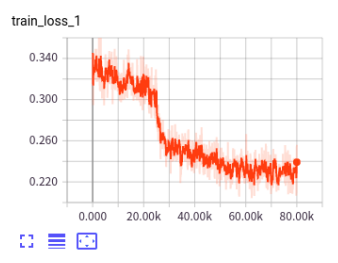
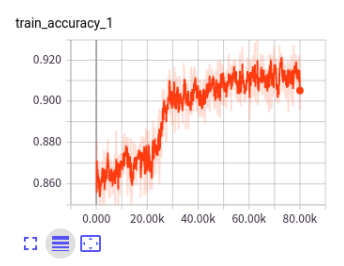
图3.4 直接训练嵌入率为0.4bpp模型accuracy和loss变化图

SRNet模型在9个训练集上训练得到的accuracy和loss变化如图3.5至3.13所示。



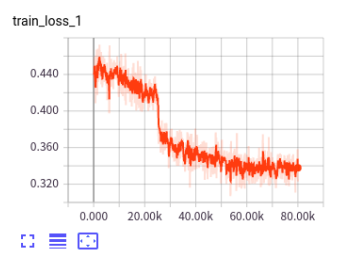
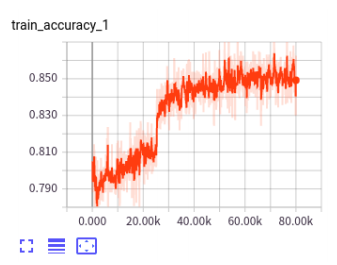
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.5 WOW隐写术1.0bpp嵌入率模型accuracy和loss变化图



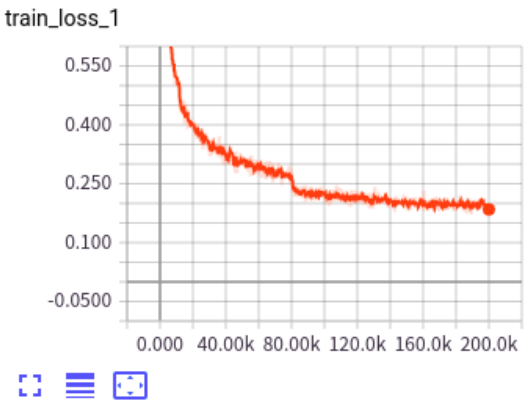
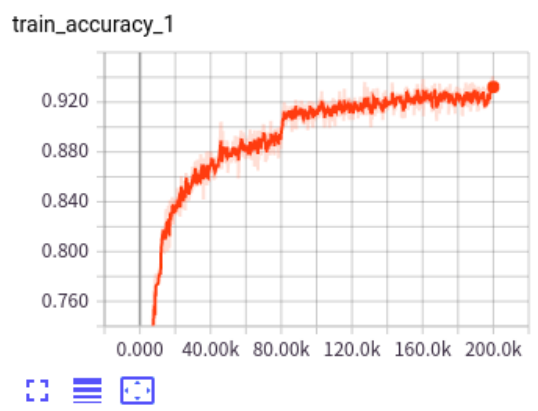
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.6 WOW隐写术0.7bpp嵌入率训练集accuracy和loss变化图



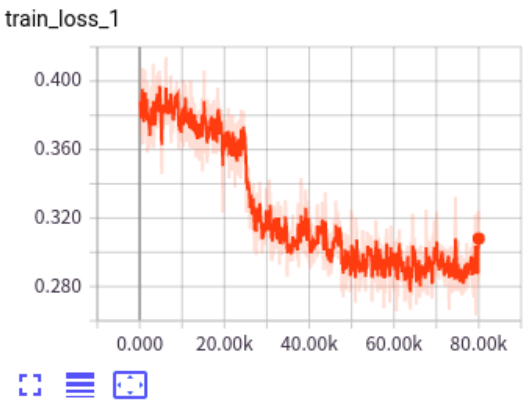
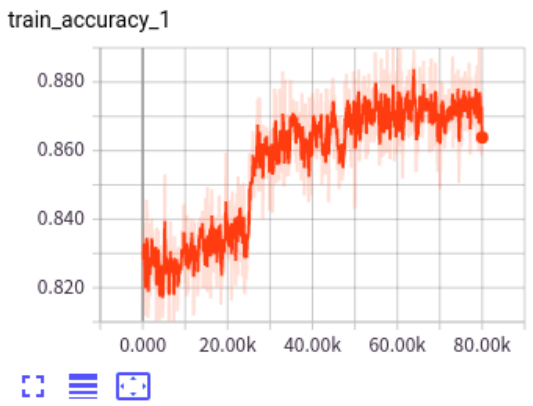
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.7 WOW隐写术0.4bpp嵌入率训练集accuracy和loss变化图



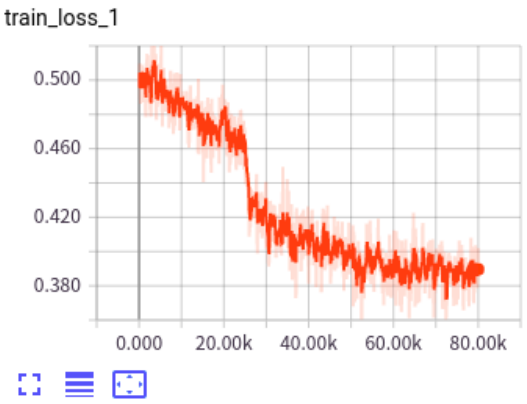
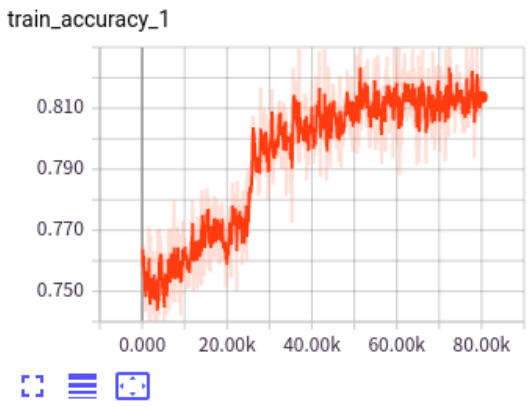
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.8 HUGO隐写术1.0bpp嵌入率训练集accuracy和loss变化图



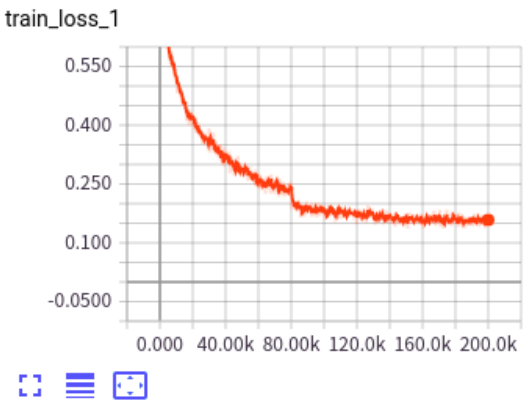
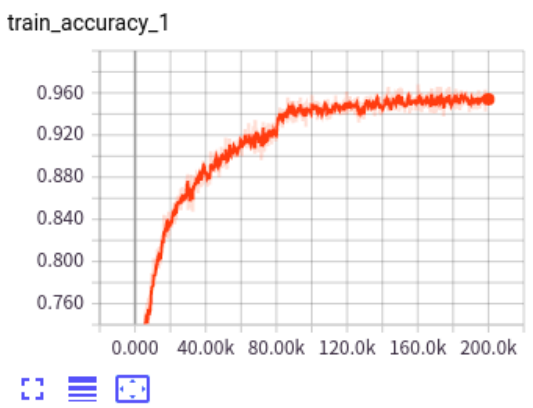
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.9 HUGO隐写术0.7bpp嵌入率训练集accuracy和loss变化图



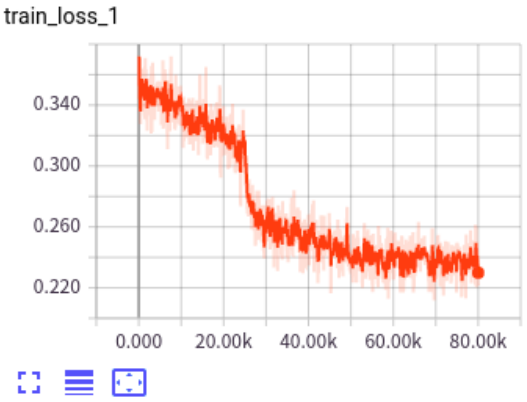
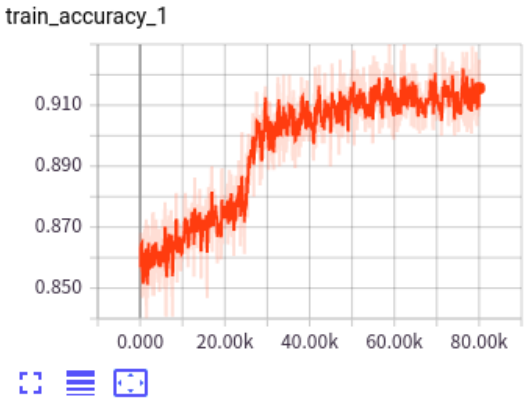
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.10 HUGO隐写术0.4bpp嵌入率训练集accuracy和loss变化图



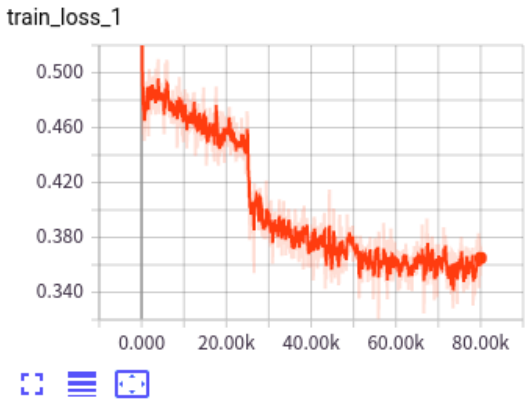
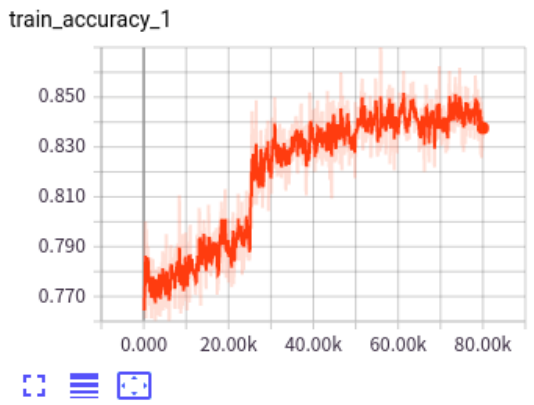
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.11 S-UNIWARD隐写术1.0bpp嵌入率训练集accuracy和loss变化图



（a）训练集accuracy变化图 （b）训练集loss变化图

图3.12 S-UNIWARD隐写术0.7bpp嵌入率训练集accuracy和loss变化图



（a）训练集accuracy变化图 （b）训练集loss变化图

图3.13 S-UNIWARD隐写术0.4bpp嵌入率训练集accuracy和loss变化图

观察图3.5至3.13可知，原始SRNet模型在由WOW、S-UNIWARD、HUGO三种隐写术以及0.4bpp、0.7bpp、1.0bpp三种嵌入率下生成的9个数据集上均可以较好的收敛。

### 3.1.2 模型测试

隐写分析领域通常采用正确率来评价模型的性能，而计算正确率首先需要获取模型在测试集上的混淆矩阵，本文仅举一例介绍混淆矩阵的格式以及准确率计算公式，模型在WOW隐写术嵌入率为1.0bpp数据集上测试的混淆矩阵如表3.1所示。

表3.1 WOW隐写术嵌入率为1.0bpp模型混淆矩阵

|  |  |  |  |
| --- | --- | --- | --- |
| 混淆矩阵 | | 预测值 | |
| Positive | Negative |
| 判定值 | True | 434 | 500 |
| False | 66 | 0 |

混淆矩阵中预测值Positive、Negative表示模型预测图像的类别，判定值True、False表示模型预测是否正确，例如表3.1中（True-Positive）值为434，这表示有434张图片其真实类别是含密图像，且模型预测为含密图像，模型预测正确，的值为66，表示模型认为有66张图片类别是含密图片，但其实是不含隐秘信息的原始图片，模型预测错误，和的涵义依此类推。在计算二分类问题准确率时，用模型预测正确的图片数量除以图片总数量即可算得模型的准确率。其计算公式如（3-1）所示。

 （3-1）

根据混淆矩阵可以计算出模型在测试集上的准确率，由于数据集太多，本文以表格的形式展示9个模型在各自测试集上的准确率，具体数据如表3.2至表3.4所示。

表3.2 嵌入率为1.0bpp各模型准确率

|  |  |  |  |
| --- | --- | --- | --- |
| 隐写术 | WOW | S-UNIWARD | HUGO |
| 准确率 | 94.08% | 93.70% | 95.54% |

表3.3 嵌入率为0.7bpp各模型准确率

|  |  |  |  |
| --- | --- | --- | --- |
| 隐写术 | WOW | S-UNIWARD | HUGO |
| 准确率 | 92.55% | 91.24% | 92.80% |

表3.4 嵌入率为0.4bpp各模型准确率

|  |  |  |  |
| --- | --- | --- | --- |
| 隐写术 | WOW | S-UNIWARD | HUGO |
| 准确率 | 87.10% | 78.30% | 88.51% |

为了更好的说明模型性能，将SRNet模型和另一个网络模型El Net[34]进行准确率对比，具体内容见表3.5。

表3.5 El Net和SRNet在HUGO数据集上准确率对比

|  |  |  |
| --- | --- | --- |
| 嵌入率 | El Net | SRNet |
| 1.0bpp | 94.3% | 95.54% |
| 0.7bpp | 90.4% | 92.80% |
| 0.4bpp | 81.7% | 88.51% |

从表3.5中可以看出SRNet模型在三种嵌入率下的准确率都高于El Net模型。

## 3.2 添加CBAM注意力机制后的网络模型

### 3.2.1 CBAM注意力机制

注意力机制在深度学习中现在已经成为一个重要的领域，注意力机制Attention Module启发于人类自身的注意力机制，神经网络在学习时，会额外关注对最终结果有帮助的特征。注意力机制早在上个世纪90年代就已经被提出，但真正受到人们重视是由于Google Mind 团队发表的Recurrent Models of Visual Attention[35]论文，由此注意力机制获得不断发展，并在自然语言处理领域取得傲人的成绩，随后有学者尝试将注意力机制引入计算机视觉领域，并提出了诸如SENet [36](Squeeze-and-Excitation Networks)、CBAM等注意力机制。本文使用的注意力机制就是CBAM机制。

卷积模块注意力模型是一种既包含通道注意力机制又包含空间注意力机制的注意力机制，相比于SENet网络只包含空间注意力机制，CBAM注意力机制的性能更加优越。

#### 3.2.1.1 通道注意力机制

在卷积神经网络中，卷积输出的特征图是卷积核与输入图像进行卷积计算后得到的结果，故输出特征图的数量（也就是输出通道数）和卷积核的数量是一样的。换句话来说，一个特征图对应一个卷积核。由于卷积核参数的不同，不同的卷积核关注的特征也不同，比如有的卷积核关注的是边缘特征；有的卷积核关注条纹状的特征；有的卷积核关注的可能恰好就是隐写嵌入时所引入的特征。这些特征有的对模型最终分析判断贡献不大，但是有的帮助却很大。在没有引入通道注意力之前，卷积输出的所有特征通道都是平等的，有相同的权重。但是引入通道注意力之后不同的通道可以被赋予不同的权重值，对于分类任务所关心的特征通道给予较大的权重，对于分类任务不关心的特征通道给予较小的权重。这样既可以增大相关特征对最终结果的贡献，又可以减少无关特征对模型的影响。

CBAM中通道注意力机制结构如图3.14所示。

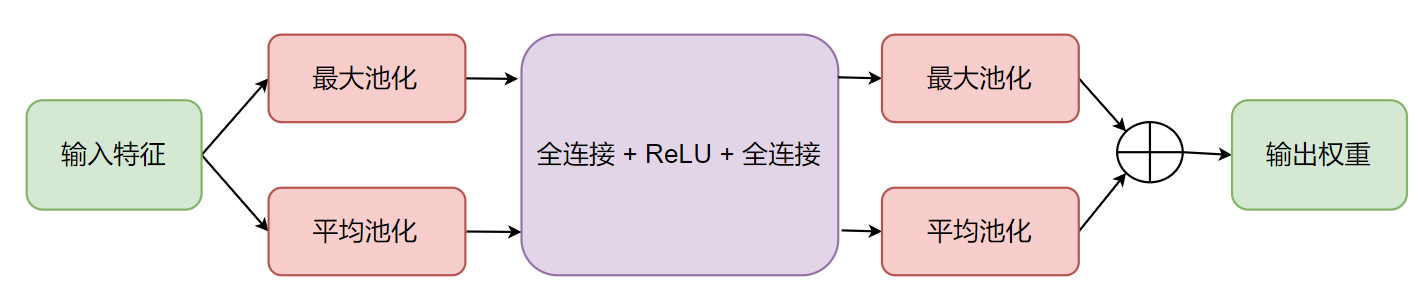


图3.14 通道注意力模块结构示意图

如图3.14所示，将上层输出的feature maps作为输入，之后将输入兵分两路，分别用最大池化和平均池化从空间上对输入进行压缩，再将压缩后的结果输入一个共享的全连接网络，之后将两者相加，再经过Sigmoid激活函数即得到最终的输出，最终输出的是一个权重矩阵，计算过程如公式（3-2）所示。

 （3-2）

其中表示输入的feature map，表示通道注意力模块的输出，表示Sigmoid激活函数、表示共享全连接结构（该全连接结构由全连接层加上ReLU激活函数再加上全连接结构组成）、和分别表示平均池化和最大池化。

最后使用Sigmoid激活函数是为了使模型输出的值在0-1之间，将得到的权重矩阵和之前的输入相乘即得到通道加权之后的feature map。

#### 3.2.1.2 空间注意力机制

通道注意力机制关注的是通道，而空间注意力关注的则是空间位置。隐写术在对图像进行秘密信息嵌入时，不可能对图片中所有的位置都进行修改，而是按照图像的内容纹理，基于内容进行自适应隐写（这样可以减少隐写前后图像的视觉差异）。所以隐写噪声特征会分布在某些通道的某些位置，如果给予这些位置的特征较大的权重，而其他位置的无关特征给予较小的权重，这样模型可以更好的提取隐写特征，从而提高模型的准确率。

通道注意力机制示意图如图3.15所示。

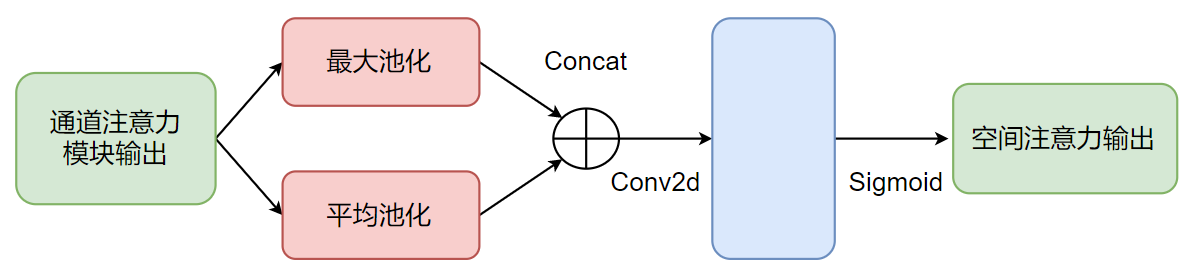


图3.15 空间注意力模块结构示意图

如图3.15所示，空间注意力模块的输入是前面通道注意力模块的输出，输入的feature map一方面在通道维度上计算其最大值，得到一个和特征图等高等宽的单通道特征，另一方面在通道维度上计算平均值，同样得到一个特征图相同高宽的单通道特征，之后将这两个特征进行Concat拼接（注意是拼接而不是相加），再经过一个卷积核为7×7大小的卷积层，最后通过Sigmoid激活函数即可得到空间注意力机制的权重矩阵。其计算流程如公式（3-3）所示。

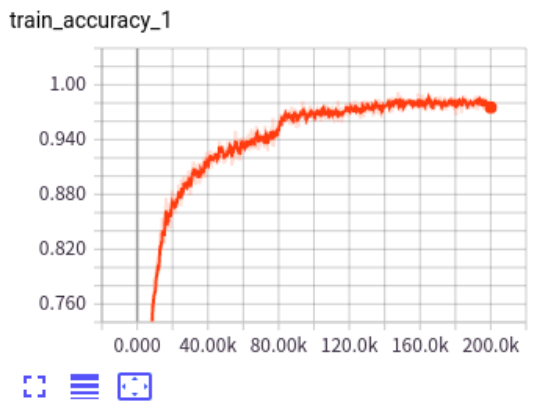
 （3-3）

其中自变量表示输入，表示空间注意力模块的输出，表示进行单核大小为7×7的卷积、表示拼接操作、和分别表示平均池化和最大池化。空间注意力模型最终输出的是一个和输入特征等高等宽的矩阵，矩阵中的值在0-1之间。最后将得到的权重矩阵和输入相乘即得到最终的输出。

### 3.2.2 模型训练

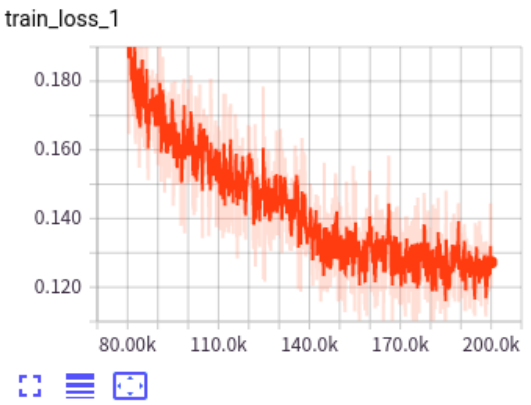
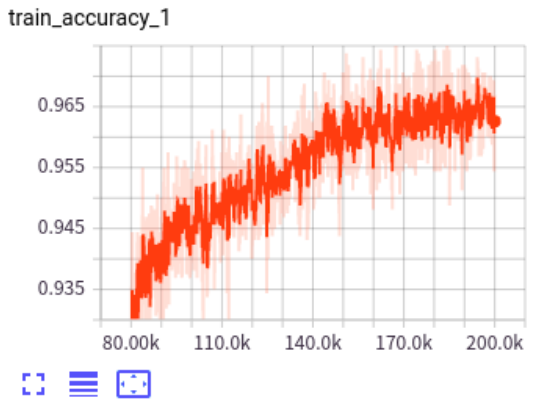
将CBAM注意力机制添加到原始SRNet模型中，由于注意力机制的主要功能是帮助模型更好的进行特征提取，而原始SRNet模型中，最重要的就是前7层进行高频噪声残差的提取，因此本文在原始SRNet网络前7层每一层的最后都额外增加了一层CBAM注意力模块。

改进后模型诸如学习率变化边界，iterations迭代次数等训练参数和改进前一致，此处不再赘述。改进后模型的训练图像如图3.16、3.17、3.18所示。其中图3.16是改进后模型在WOW隐写术，嵌入率为1.0bpp的数据集上训练得到的loss和accuracy变化图；图3.17对应HUGO隐写术，嵌入率为1.0bpp；图3.18对应S-UNIWARD隐写术，嵌入率为1.0bpp。



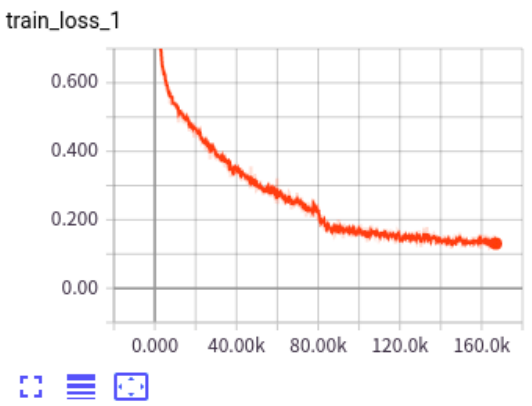
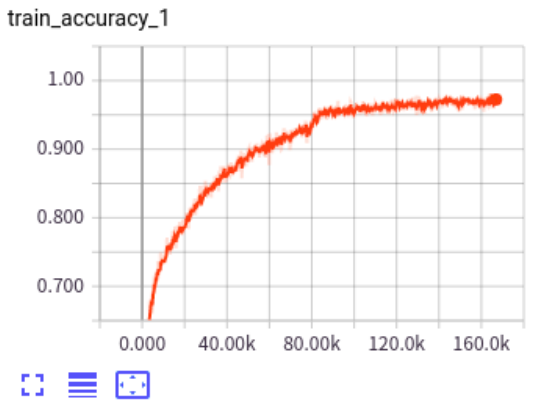
（a）训练集accuracy变化图 （b）训练集loss变化图

图3.16 优化后WOW-1.0bpp嵌入率训练集accuracy和loss变化图



（a）训练集accuracy变化图 （b）训练集loss变化图

图3.17 优化后S-UNIWARD-1.0bpp训练集accuracy和loss变化图



（a）训练集accuracy变化图 （b）训练集loss变化图

图3.18 优化后WOW-1.0bpp嵌入率训练集accuracy和loss变化图

观察图3.16、3.17、3.18可知，改进后模型在嵌入率为1.0bpp，WOW、HUGO、S-UNIWARD三种隐写术上均可以较好的收敛。

### 3.2.3 模型测试与结果对比分析

按照公式（3-1）计算改进后模型的准确率，其具体准确率如表3.6所示，对应直方图如图3.19所示。其中横坐标表示三种隐写术、纵坐标表示嵌入率为1.0bpp时原始SRNet模型和改进后的CBAM-SRNet模型在测试集上的准确率，并且蓝色表示原始SRNet模型，黄色表示改进后的CBAM-SRNet模型。

表3.6 SRNet和CBAM-SRNet嵌入率为1.0bpp准确率对比

|  |  |  |
| --- | --- | --- |
| 隐写术 | SRNet | CBAM-SRNet |
| WOW | 94.08% | 95.61% |
| HUGO | 93.70% | 94.62% |
| S-UNIWARD | 95.54% | 97.15% |

观察表3.6可知，当嵌入率为1.0bpp时，在WOW、HUGO、S-UNIWARD三种隐写术上，增加CBAM注意力机制后的模型比原始的SRNet模型由更高的准确率。这证明注意力机制对提升SRNet网络性能有帮助。

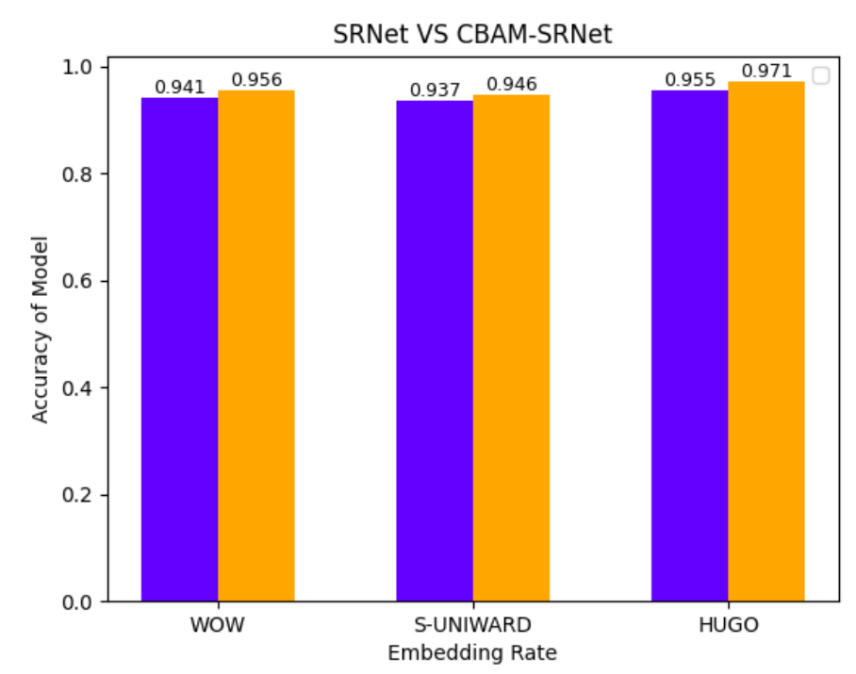


图3.19 SRNet和CBAM-SRNet模型的准确率对比

## 3.3 本章小结

本章阐述了一种基于深度学习的图像隐写分析方案SRNet以及利用CBAM注意力机制改进原始SRNet网络。本章第一节详细介绍了SRNet模型的网络结构，以及SRNet模型的训练参数和训练过程中accuracy以及loss的变化图像。第二节详细介绍了CBAM注意力机制中的空间注意力机制和通道注意力机制，以及如何将CBAM注意力机制添加到原始SRNet模型上得到改进后的模型，最后计算改进前后模型的准确率，对比发现增加CBAM注意力之后模型的准确率高于原始SRNet模型，由此可证明，增加CBAM注意力机制之后，模型的性能确实得到了提升。

# 第4章 基于生成对抗网络的图像隐写去除

虽然将深度学习应用于隐写分析领域使得隐写分析相比以往有了长足的进步，但面对同样在不断进步的隐写术，隐写分析依旧处于落后状况。并且隐写分析还受到诸如：图片大小、颜色空间、文件格式等因素的限制。所以面对隐写分析受阻的局面，本文转换思路：既然很难通过隐写分析判断图像中是否含有隐秘信息，那么不妨对有含密嫌疑的图片统统进行隐写去除操作，也就是破坏含密图像中的隐秘信息，这样即使对方接收到含密图像，由于其中的秘密信息都已经被破坏，对方也就无法知晓图片中原本嵌入的信息。目前在隐写去除领域已经提出诸如去噪滤波器等方法，这些方法虽然能够去除含密图像中的隐秘信息，但却无法保障生成图像的视觉质量。

本章基于GAN网络和自编码器实现了隐写去除，在破坏含密图像中秘密信息的同时，尽最大可能保持图像的视觉质量不变。

## 4.1 DDSP网络结构

DDSP 网络模型和SRGAN网络的结构非常类似，后者是提升图像分辨率领域非常有名的网络模型，只不过不同于SRGAN使用大型的ResNet残差网络，DDSP使用一个预训练好的自编码器作为生成器Generator，其目的是去除隐写内容的同时尽可能的保留图像质量。DDSP网络结构如图4.1所示。

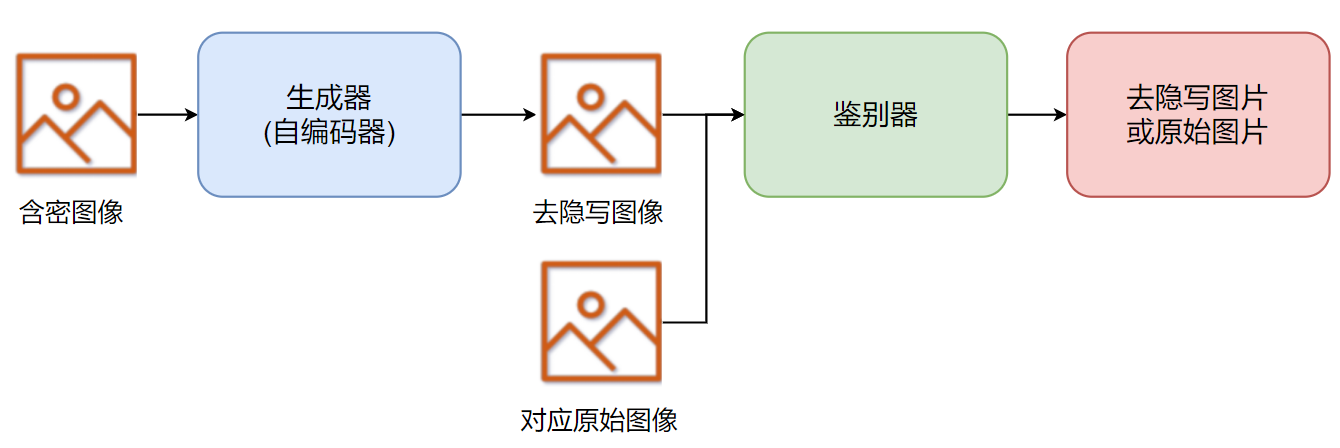


图4.1 DDSP网络结构示意图

如图4.1所示，DDSP网络是一个GAN网络，其中生成器Generator是一个自编码器，向生成器中输入含密图像，输出的就是去除含密信息后的去隐写图像，鉴别器Discriminator是一个普通的卷积神经网络，向鉴别器中输入不含隐秘信息的原始图像或Generator输出的去隐写图像，输出的是一个0-1之间的常数，该常数表示输入图像是真实图像的概率。

### 4.1.1 生成器网络结构

生成器是一个自编码器，属于无监督学习（Unsupervised learning）。其网络结构如图4.2所示。

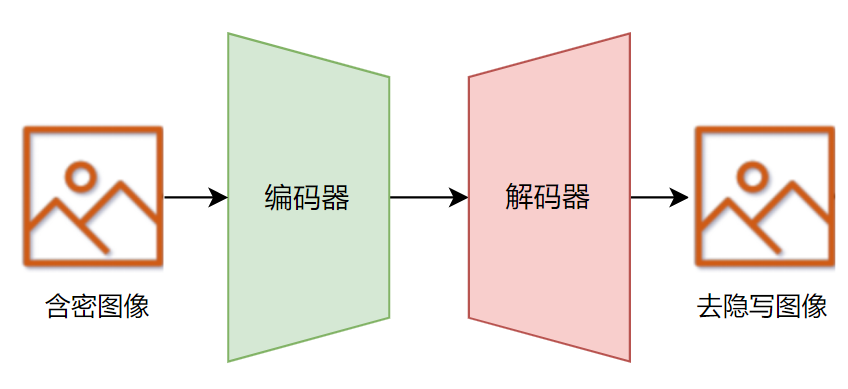


图4.2 自编码器结构示意图

自编码器包含编码器Encoder和解码器Decoder两部分，其作用简单来说就是用自身的高阶特征编码自己。编码器中的Encoder从庞杂的输入中提取特征（这些高维特征几乎可以表示输入本身，相比于原始输入更加凝练），而后将提取的特征输入到Decoder中，Decoder再根据这些特征进行复原，期望尽可能的恢复到和之前输入的状态一致。面对海量且高维的数据时，人们一般都会用自编码器进行预训练，之后将自编码器中Encoder的输出作为网络模型的输入，这样不仅可以减少不必要数据的输入，降低内存占用率，也可以使模型更容易收敛。

本文中自编码器的作用与之类似，Encoder的输入为含密图片，其作用是降低图像的大小同时尽可能多的保留图片信息，Decoder的作用是将图片恢复到原始大小，同时去除其中嵌入的秘密信息。Encoder的详细结构如图4.3所示。

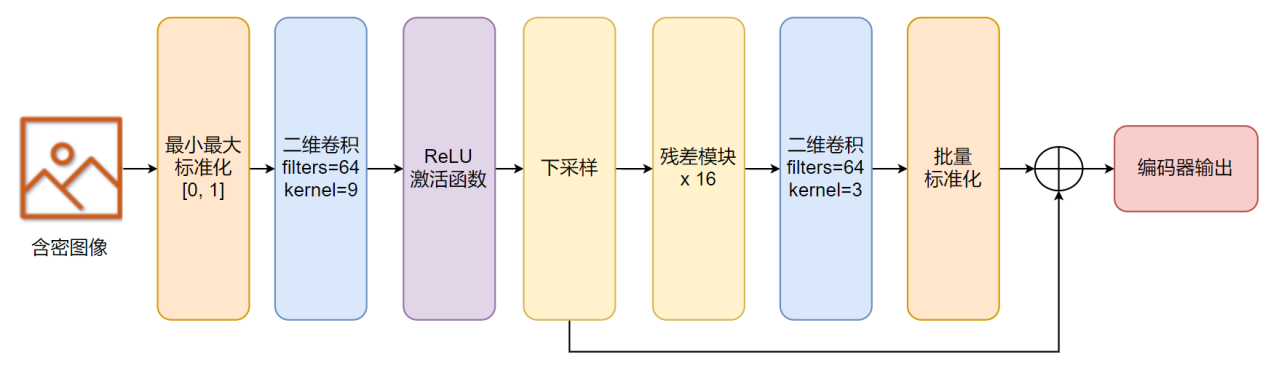


图4.3 自编码中encoder结构示意图

如图4.3所示，将图片输入到Encoder之后，首先经过最小最大标准化 (Min-Max Normalization)，其转换过程如公式（4-1）所示。

 （4-1）

在公式（4-1）中，表示像素点变换前的像素值，表示该像素点变换后的像素值，表示该图片中最小像素值，表示该图片中最大像素值。标准化之后进行卷积操作，其中有64个大小为9×9的卷积核，卷积核移动步长为1，padding等于4，卷积后的结果再经过ReLU激活函数，进入到下采样Down-Sample模块，下采样模块具体结构如图4.4所示。

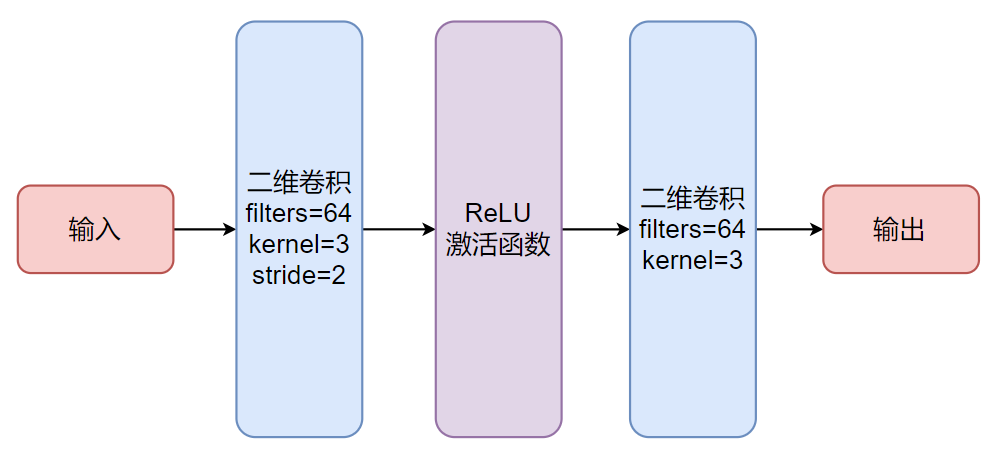


图4.4 下采样Down-Sample模块结构示意图

下采样模块的格式为卷积、ReLU、卷积的格式，其中需要注意的是第一层卷积，该卷积中采用64个大小为3×3的卷积核，且移动步长为2。正是由于卷积核移动的步长为2，所以输入图片的边长变为输入图片的一半。一般来说下采样操作都是用池化而不是卷积来实现，这里用卷积的目的是为了让模型能够学习如何使用最优的方法进行下采样，从而提高输出图像的质量。

下采样模块的输出，之后被传递进入16个串行的残差模块（Residual Module），残差模块的结构如图4.5所示。

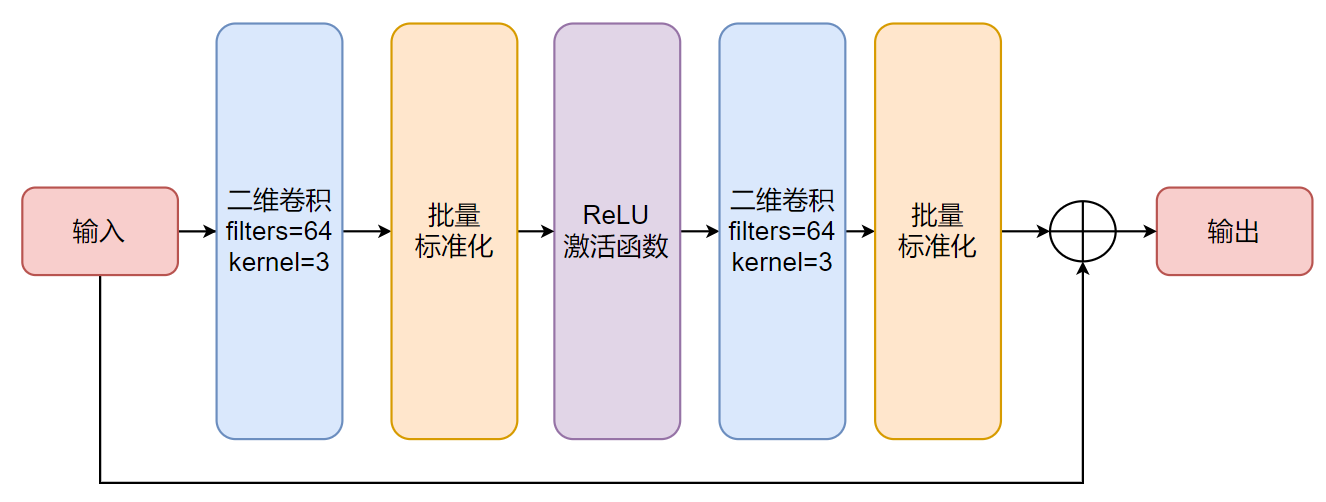


图4.5 残差模块结构示意图

残差模块之后是一个二维卷积以及批量标准化，批量正则化后的输出再加上之前下采样模块的输出，即得到Encoder最终的输出。

Encoder中卷积、池化、全连接，批量标准化层的参数详见表4.1所示。

表4.1 Encoder网络详细参数

|  |  |  |  |
| --- | --- | --- | --- |
| 网络层次 | 输入维度 | 输出维度 | 网络参数 |
| MinMaxNorm | [N, 1, 256, 256] | [N, 1, 256, 256] | None |
| Conv2D | [N, 1, 256, 256] | [N, 64, 256, 256] | 9×9, 64, stride=1, padding=4 |
| ReLU | [N, 1, 256, 256] | [N, 1, 256, 256] | Inplace = True |
| Down-Sample | [N, 1, 256, 256] | [N, 64, 128, 128] | 3×3, 64, stride=1, padding=1 |
| ResBlock × 16 | [N, 64, 128, 128] | [N, 64, 128, 128] | 3×3, 64, stride=1, padding=1 |
| Conv2D | [N, 64, 128, 128] | [N, 64, 128, 128] | 3×3, 64, stride=1, padding=1 |
| Batch Norm | [N, 64, 128, 128] | [N, 64, 128, 128] | None |

表4.1输入输出维度一列中[N, 1, 256, 256] 分别表示[batch size, channels, height, weight] 即图片数量、通道数、图片高度、图片宽度。

输入经Encoder之后即进入Decoder部分，Decoder网络结构如图4.6所示。

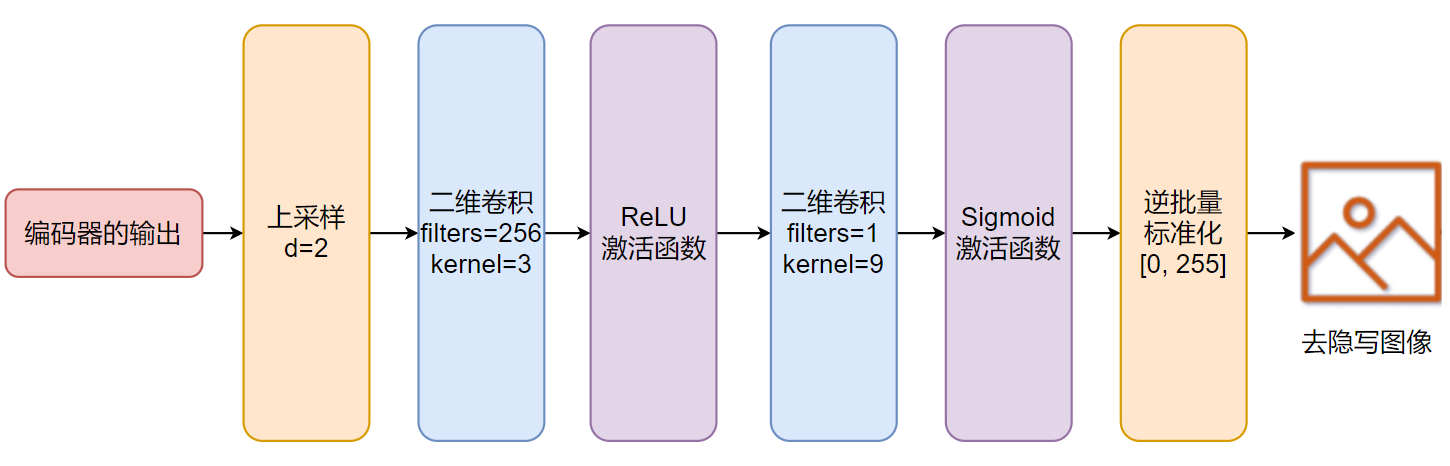


图4.6 Decoder网络结构示意图

Decoder网络的输入首先经过缩放因子为2的上采样，上采样的方式为临近插值法 (near interpolation)，这样图片就恢复成Encoder最开始输入的大小，之后紧随一个卷积操作，卷积核个数为256大小为3×3，卷积层的输出进入ReLU激活函数，再进行卷积操作，卷积核个数为1大小为9×9，再使用Tanh激活函数进行非线性变换，Tanh函数曲线类似于Sigmoid函数都是一个S型曲线，只不过Tanh函数关于原点中心对称，且值域是[-1, 1]，而Sigmoid函数不关于原点中心对称且值域在[0, 1]之间。

Tanh函数输出的张量，其值域在[-1, 1]之间，而图片数据是整型值域为[0, 255]，因此需要进行转换。本文转换的思路是将Tanh函数的输出先经MinMax Norm标准化将值域由[-1, 1]转化为[0, 1]，之后再将标准化的结果乘以255，再下取整即可得到值域为[0, 255]的图片数据。

但最终在代码实现时，发现训练出的模型完全不收敛，其loss值甚至越来越大，Debug发现问题出现在[-1, 1]到[0, 1]的变换过程中，尝试修改代码无果后只好更换网络结构，将Decoder最后一层Tanh激活函数替换为Sigmoid函数。从而绕过MinMax Normalization这一步，将Sigmoid函数输出的张量直接乘以255即可。

Decoder中网络层参数详见表4.2所示。

表4.2 Decoder网络详细参数

|  |  |  |  |
| --- | --- | --- | --- |
| 网络层次 | 输入维度 | 输出维度 | 网络参数 |
| Upsample | [N, 64, 128, 128] | [N, 64, 256, 256] | Scale factor = 2 |
| Conv2D | [N, 64, 256, 256] | [N, 256, 256, 256] | 3×3, 256, stride=1, padding=1 |
| ReLU | [N, 256, 256, 256] | [N, 256, 256, 256] | Inplace = True |
| Conv2D | [N, 256, 256, 256] | [N, 1, 256, 256] | 9×9, 1, stride=1, padding=4 |
| Sigmoid | [N, 1, 256, 256] | [N, 1, 256, 256] | None |
| DeNorm | [N, 1, 256, 256] | [N, 1, 256, 256] | [0, 1]→[0, 255] |

### 4.1.2 鉴别器网络结构

生成器其实已经完成了本文预期目标，即去除含密图像中的秘密信息。但之所以还要在自编码器的外面增加一层GAN网络的框架，将自编码器作为GAN网络中的Generator同时添加鉴别器Discriminator部分，其目的就是进一步对自编码器进行训练，保证完成隐写去除的基础上，使输出的图像质量更高，和不包含秘密信息的原始图片在视觉效果上更加接近。

Discriminator的网络结构如图4.7所示。

将Generator输出的去隐写后的图片和不包含水印信息的原始图像输入到Discriminator网络中，首先经过最大绝对值标准化（MaxAbs Normalization），其计算过程如公式（4-2）所示。

 （4-2）

在公式（4-2）中，表示张量中某点变换前的值，表示对应点变换后的值，表示将张量中所有的值先求绝对值然后求最大值。输入图像中每一个像素点的像素值值域为[0, 255]，经最大绝对值标准化之后，值域变为[-1, 1],标准化后的输出首先进行卷积操作，卷积核大小为3×3，个数为512，步长为1，卷积之后再经过ReLU激活函数，随后进入7个结构相似的Block模块中，每一个Discriminator Block模块卷积核大小都是3×3，移动步长和卷积核个数各不相同，Block模块具体结构如图4.8所示。依次经过7个Block模块后，进行Flatten操作，将张量拉伸为一个二维矩阵，矩阵中的每一行对应之前的一个C×H×W特征图，随后再经过包含1024个神经元的全连接层、ReLU激活函数、以及1个神经元的全连接层，最后经过Sigmoid激活函数（原网络结构中为Tanh激活函数）得到一个常数，该常数表示输入图像是真实图像（不含隐秘信息的原始图像）的概率。

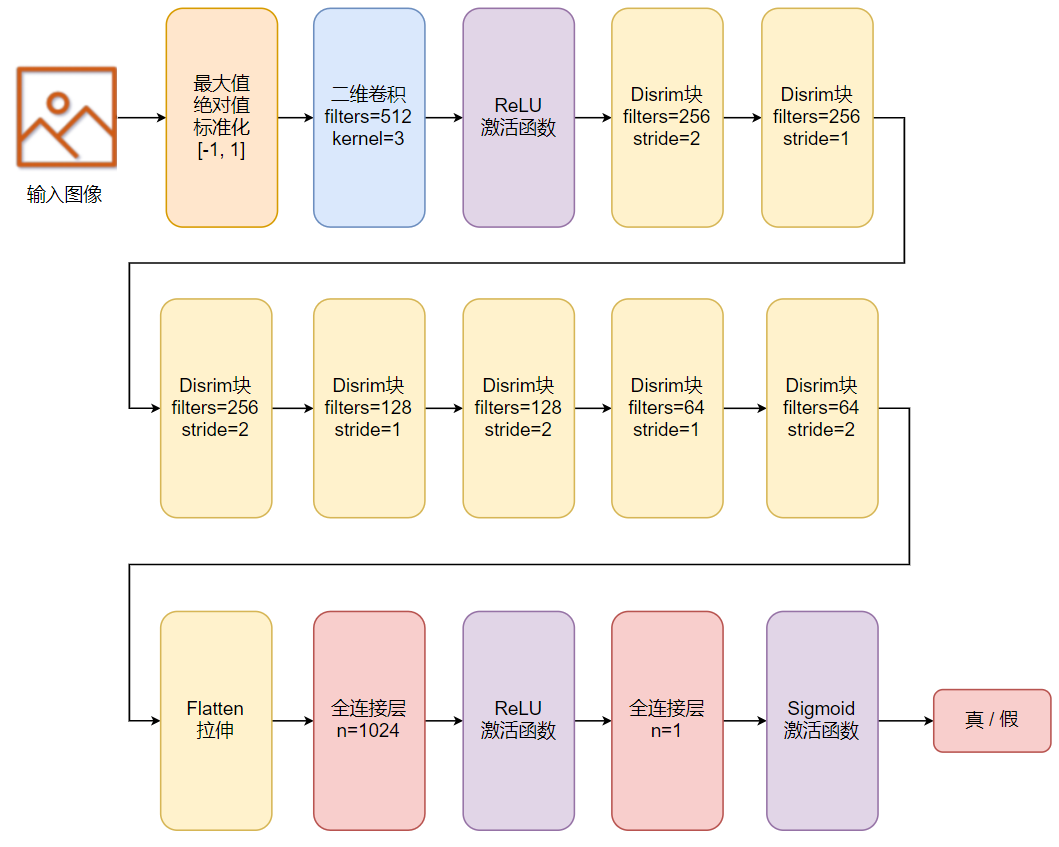


图4.7 鉴别器网络结构示意图

DDSP网络模型中Discriminator的网络结构和SRGAN网络模型中Discriminator的网络结构很类似，只是Discriminator Block的结构不同，其他部分都一致。相比于SRGAN网络，DDSP网络中Block模块卷积操作使用的卷积核更少，结构较为简单，所以模型训练所需时间也更少。

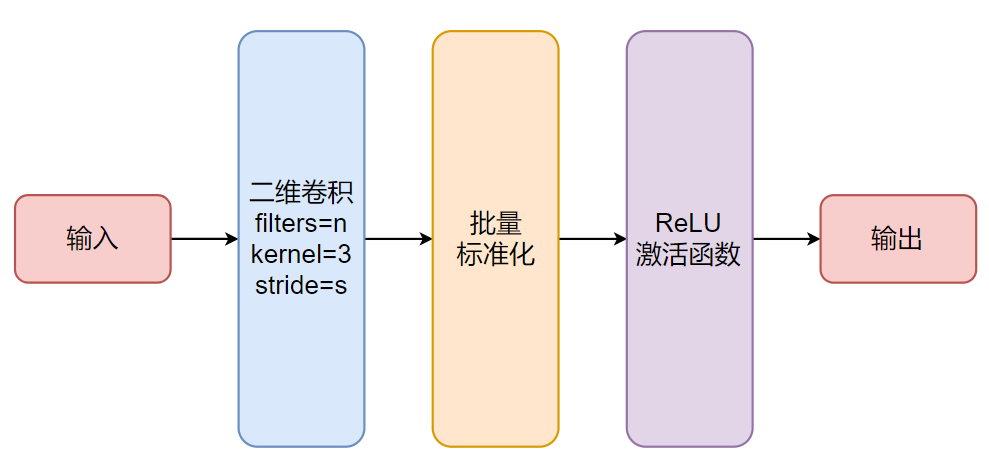


图4.8 鉴别器中卷积模块结构示意图

如图4.8所示，鉴别器中卷积模块中只有三个网络层，第一层是卷积操作，卷积核大小均为3×3；第二层是批量标准化Batch Normalization；第三层是ReLU激活函数。

鉴别器Discriminator中网络层参数详见表4.3所示。

表4.3 鉴别器网络详细参数

|  |  |  |  |
| --- | --- | --- | --- |
| 网络层次 | 输入维度 | 输出维度 | 网络参数 |
| MaxAbs Norm | [N, 1, 256, 256] | [N, 1, 256, 256] | None |
| Conv2D | [N, 1, 256, 256] | [N,512,256,256] | 3×3, 512, stride=1, padding=1 |
| ReLU | [N,512,256,256] | [N,512,256,256] | Inplace=True |
| Discrim Block1 | [N,512,256,256] | [N,512,128,128] | 3×3, 512, stride=2, padding=1 |
| Discrim Block2 | [N,512,128,128] | [N,256,128,128] | 3×3, 256, stride=1, padding=1 |
| Discrim Block3 | [N,256,128,128] | [N, 256, 64, 64] | 3×3, 256, stride=2, padding=1 |
| Discrim Block4 | [N, 256, 64, 64] | [N, 128, 64, 64] | 3×3, 128, stride=1, padding=1 |
| Discrim Block5 | [N, 128, 64, 64] | [N, 128, 32, 32] | 3×3, 128, stride=2, padding=1 |
| Discrim Block6 | [N, 128, 32, 32] | [N, 64, 32, 32] | 3×3, 64, stride=1, padding=1 |
| Discrim Block7 | [N, 64, 32, 32] | [N, 64, 16, 16] | 3×3, 64, stride=2, padding=1 |
| Flatten | [N, 64, 16, 16] | [N, 64 × 16 × 16] | None |
| Fully Connect | [N, 16384] | [N, 1024] | [64 × 16 × 16, 1024] |
| ReLU | [N, 1024] | [N, 1024] | Inplace=True |
| Fully Connect | [N, 1024] | [N, 1] | [1024, 1] |
| Sigmoid | [N, 1] | [N, 1] | None |

总的来说，DDSP网络是一个和SRGAN网络结构类似的网络模型，其中Generator是一个自编码器，Discriminator是一个普通的卷积神经网络。Generator输入含密图像、输出去除隐写信息后的图像；Discriminator输入的是不含有秘密信息的原始图像和Generator输出的隐写去除后的图片，输出为常数，该常数表示输入图片是原始图片的概率。身为Generator的自编码器本身其实已经实现了隐写去除的任务，之所以要增加GAN的训练框架，是为了进一步对自编码器进行训练，使其输出的图像质量更高，和原始图像在视觉效果上更加接近。

## 4.2 模型训练

### 4.2.1 训练参数设置

模型训练需要分步训练，先训练自编码器，自编码器收敛之后，再代入到GAN网络框架中继续训练，这样可以进一步提升模型生成图片的视觉质量。

自编码器输入的图片是由HUGO隐写术，嵌入率为1.0bpp进行隐写信息嵌入而生成的大小为256×256的灰度图。数据集的大小与划分和隐写分析数据集一致，均为20000张图片，其中训练集14000张、验证集1000张、测试集5000张。在训练超参数方面，自编码器训练采用的损失函数为均方差损失函数；优化器为Adam(β1=0.5、β2=0.9)；数据集遍历次数epochs为20；学习率初始化为1e-3，当epoch等于8或12时，学习率减少为原来的十分之一，也就是1e-4和1e-5；训练的batch size为14，验证集上的batch size为10。

 （4-3）

自编码器训练收敛之后，开始训练GAN网络。

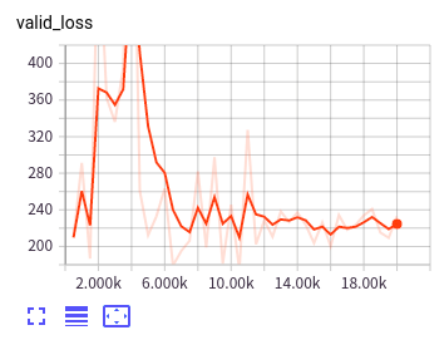
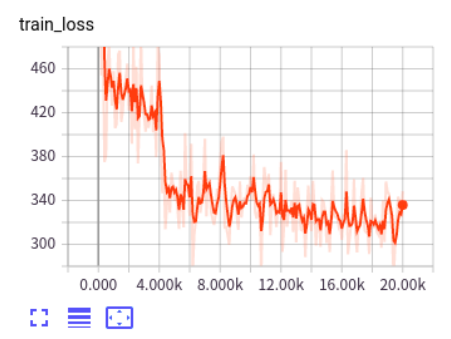
其中Generator自编码器训练方式和单独训练时保持一致，此处不再赘述。鉴别器Discriminator训练采用的优化器为Adam，损失函数为均方差损失函数。此外由于网络结构复杂且显卡容量有限，将训练集和验证集的batch size降低为原来的一半，即训练的batch size由之前的14减少为7，验证的batch size由之前的10减少为5。GAN网络训练的epochs数为8个epoch，学习率一开始设置为1e-4，当epoch=4时变为1e-5。

GAN网络训练过程中值得一提的是损失函数的设计。最初DDSP模型采用的损失函数是原始GAN网络损失函数（如公式（4-3）所示），但在训练时出现无穷小NAN错误，这和原始GAN网络损失函数中的log计算有关，当函数输入接近0时，log函数的输出会逼近无穷小从而导致下溢。于是舍弃原始GAN网络损失函数，改用均方差损失函数。具体方法为：设置两个target label，一个是内容全1的矩阵，其形状为[batch\_size, 1]，另一个是相同形状的全0矩阵，对于Discriminator而言，当输入为Generator生成的假图片时，设置其label为全0矩阵；当输入为不包含隐藏信息的原始图片时，设置其label为全1矩阵，求出两者均方差，加和除2，得到的值作为Discriminator最终的损失值。对于Generator而言，其损失值一方面来自于生成图像和含密图像之间的均方差损失（含密图像和去隐写图像对应点的像素值相减，求其平方值，并加和求平均值），另一方面来自GAN网络损失，将Generator网络生成的图像输入到Discriminator后，设置其label为全1矩阵，求出其均方差，再乘以1e-3。两个均方差损失值求和所得的值作为Generator最终的损失值。

采用均方差损失做为损失函数值时，一开始Generator的损失值没有包含生成图像和生成图像的均方差这一项，导致添加GAN网络训练之后生成图像的视觉质量反而越来越差，添加补充后，模型训练才正常。

### 4.2.2 训练输出

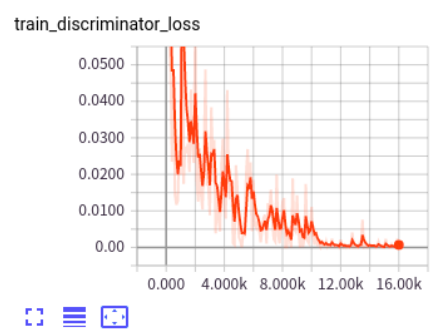
自编码器训练输出如图4.9所示，其中左图是训练集上的损失值下降图，右图是在验证集上测试得出的损失值下降图。



（a）训练集loss变化示意图 （b）验证集loss变化示意图

图4.9 自编码器loss变化图

将自编码器代入到GAN网络框架中继续训练，其输出如图4.10所示，其中左图为鉴别器的损失变化图、右图是生成器的损失变化图。

（a）鉴别器loss变化示意图 （b）生成器集loss变化示意图

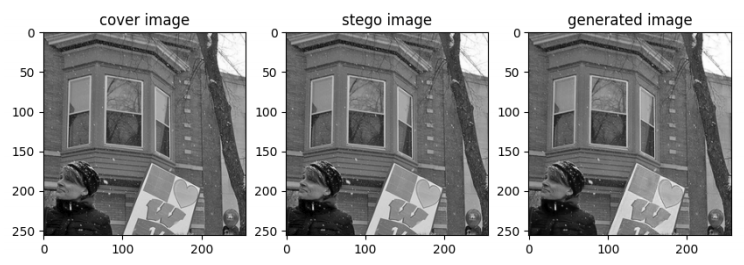
图4.10 GAN网络训练loss变化图

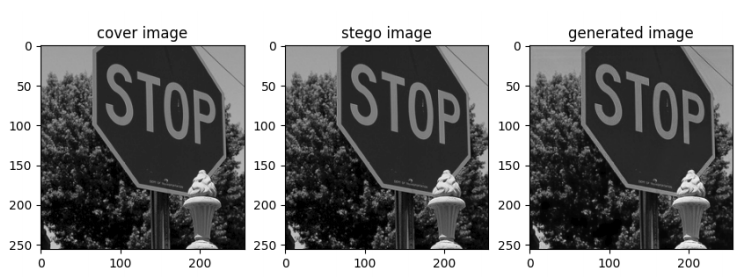
观察图4.10自编码器和GAN网络训练loss的变化图可知，模型正常收敛，且自编码器在GAN框架下训练后，其loss值进一步下降（见图4.10右图），这说明增加对抗训练之后模型的性能有了进一步提高。

## 4.3 模型测试与结果分析

DDSP模型任务是去除含密图像中的秘密信息，同时尽可能保证图像的视觉质量不受影响。模型训练完毕后，向模型中输入含密图像，查看输出图像视觉质量，部分输出如图4.11所示。

图4.11中左侧图像是不含秘密信息的原始图像、中间图像是含密图像，右侧图像是DDSP模型输出的隐写去除后的图像。从肉眼上来看三者没有明显区别，这表明生成的去隐写图像从主观上看具有较好的视觉效果。





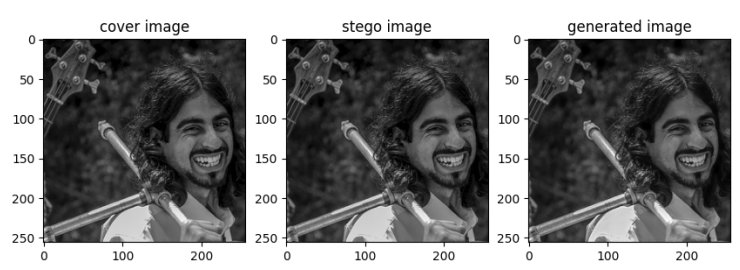


图4.11 DDSP模型去隐写后的图像

本文用误码率（Bit Error Rate，BER）来衡量模型隐写去除的好坏，误码率越大，证明模型生成的含密图像相比于输入的含密图像的改动越大（保证图像质量的前提下），图像中嵌入的隐秘信息受到的破坏也就越大。由于测试集较大，每次测试本文均从测试集中随机选择10张图片，计算这些图像的误码率BER、均方差（Mean Square Error, MSE）、标准差（Standard Deviation SD）、峰值信噪比[37] (Peak SigNal to Ratio，PSNR)、结构相似性[38] (Structual SIMilarity index，SSIM)。

测试结果具体内容如表4.4所示。

表4.4 测试结果示意图

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 测试序号 | 误码率 | 均方差 | 标准差 | 峰值信噪比 | 结构相似性 |
| 1 | 68.98% | 54.45 | 6.19 | 31.65 | 96.92% |
| 2 | 67.94% | 83.54 | 8.40 | 30.19 | 97.59% |
| 3 | 69.56% | 53.14 | 6.95 | 30.76 | 96.66% |

观察表4.4可知，误码率均达到65%以上，这说明含密图像中超过65%的比特位发生了变化，含密图像中的隐秘信息受到很大的修改和破坏，成功实现了隐写去除任务。

此外由表中峰值信噪比以及结构相似性可知，生成的去隐写图像和网络输入的含密图像在视觉质量上很接近，这也从实验数据方面证明模型在进行隐写去除的同时，能够保证输出图像具有较好的图像质量。

此外，在自编码器的基础上增加GAN对抗训练之后，生成器的性能获得提升并且生成图像的质量也有进一步提高，具体情况如图4.12所示。

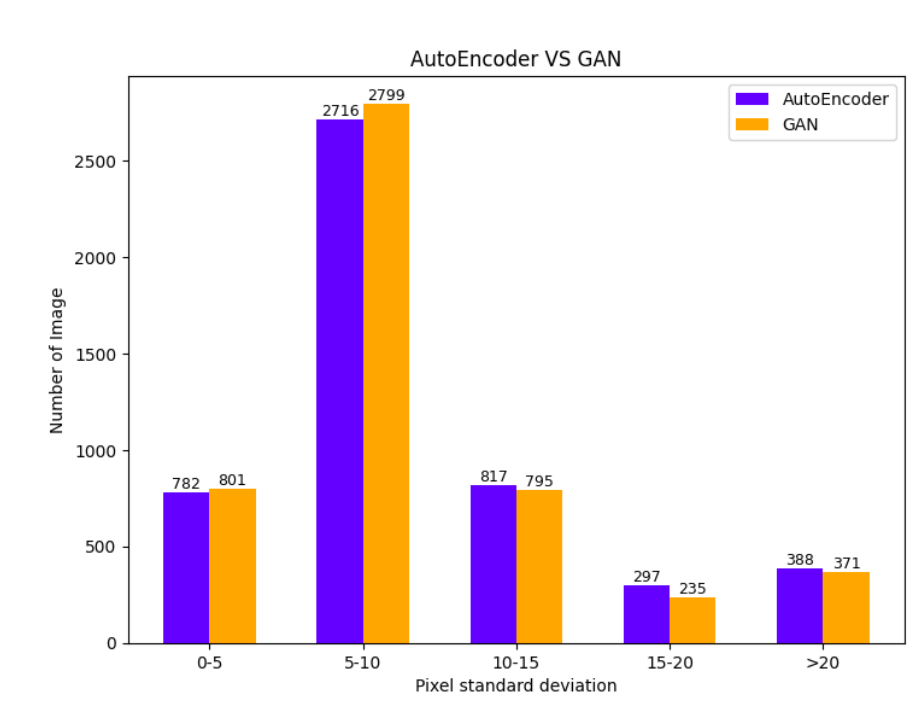


图4.12 增加GAN对抗训练后模型性能提升示意图

图4.12中横坐标表示图像像素值标准差（求含密图像和去隐写图像之间的平均标准差，例如含密图像中某一点的像素值为100，则0-5表示生成的去隐写图像对应位置的像素值在100±5之间），纵坐标表示像素差值在误差范围内的图像的数量，蓝色表示仅训练自编码器的测试结果，黄色表示增加GAN对抗训练后的模型测试结果。

观察图4.12发现，增加GAN对抗训练之后，误差较小的图像数量增多了（像素差值在0-5之间的图像数量从782上升到了801），误差较大的图像数量减少了（像素差值在15-20之间的图像数量从297下降到了235）。这证明，增加GAN网络对抗训练框架之后模型的性能获得了进一步提升。

## 4.4 本章小结

本章实现了一个基于深度学习的图像隐写去除网络模型DDSP。

首先详细阐述了DDSP模型中生成器Generator和鉴别器Discriminator网络的网络结构，其中生成器是一个自编码器，鉴别器是一个普通的卷积神经网络。之后介绍了模型训练过程中超参数的设置，包括学习率、损失函数、优化器、batch size等等，同时还分别展示了自编码器和GAN网络模型的loss变化图像。

本章最后展示了模型生成的去隐写图像，通过计算图像的误码率BER、均方差MSE、标准差SD、峰值信噪比PSNR、结构相似性SSIM，证明模型能够实现隐写信息去除，并保证隐写去除后图像依旧具有较好的视觉质量。最后通过对比直接训练自编码器和增加GAN生成对抗训练两种训练方式，证明增加GAN对抗训练之后，模型的性能得到进一步提升。

# 第5章 可视化演示系统设计

为了更直接的展示模型隐写嵌入和隐写分析的过程，实验最后用PyQt5制作可视化界面，演示隐写图像隐写和隐写分析操作。

## 5.1 登录注册界面

运行程序，出现如图5.1所示的界面。



图5.1 演示系统首页面

点击用户注册，输入注册号、姓名、密码后点击注册按钮即可实现用户注册、点击用户登录按钮，输入注册号和密码，即可实现用户登录。用户注册和登录界面如图5.2所示。

（a）用户注册界面 （b）用户登录界面

图5.2 用户注册和登录界面

在登录和注册时，后端会在数据库用户信息表中查询核对相关信息，若出现诸如用户注册号已注册或者登录密码错误等问题，将会提示用户重新操作。

## 5.2 隐写嵌入和隐写分析界面

用户成功登录后就会跳转到隐写嵌入和隐写分析主界面，如图5.3所示。界面上半部分进行隐写嵌入，下半部分进行隐写分析。

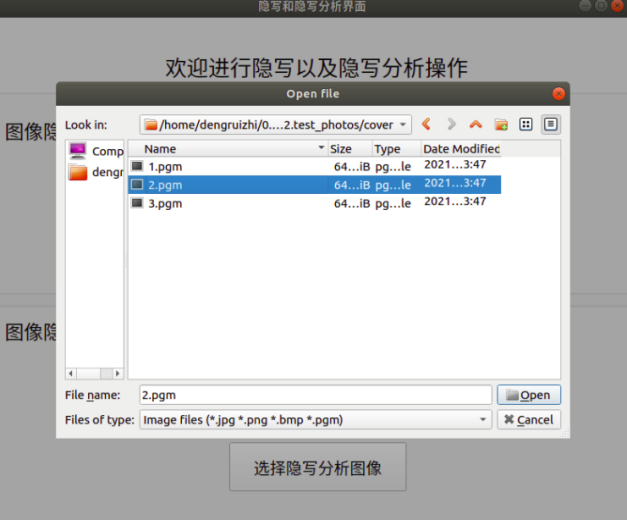


图5.3 隐写嵌入和隐写分析主界面

如图5.3所示，进行隐写嵌入时，分别点击隐写术和嵌入率对应的下拉框，其中隐写术有WOW、HUGO、S-UNIWARD三类，嵌入率有0.4bpp、0.7bpp、1.0bpp三类。选择好想要的隐写术和嵌入率之后点击确认，弹出信息框，确认选择无误之后点击确认，将会弹出文件夹，双击进行嵌入的原始图片即可实现隐写嵌入。其对应界面如图5.4、5.5所示。



图5.4 信息选择确认界面

（a）选择图片隐写 （b）嵌入成功后界面弹窗

图5.5 图片选择及隐写嵌入界面

隐写嵌入之后，会将生成的含密图像保存到指定目录中，等待隐写分析。

图5.3下半部分是图像隐写分析界面，其界面分布同上半部分的隐写嵌入界面一致。用户在进行隐写分析时，首先需要从下拉框中选择嵌入率和隐写术，之后系统后端会根据选择的隐写术和嵌入率确定需要加载的模型参数文件，这里的模型参数文件就是之前第3章中SRNet网络按照不同的隐写术和嵌入率训练好的文件。点击“选择隐写分析图像”按钮后，从弹出的页面中选择需要隐写分析的图片，双击选中图片，系统读取该图片并将其输入SRNet网络模型，利用训练好的模型参数文件初始化网络模型，开始进行图像隐写分析，并通过弹窗的形式告知模型判断结果。

检测结果为原始图像和含密图像，不同的弹窗如图5.6、5.7所示。



图5.6 检测结果为原始图像界面



图5.7 检测结果为含密图像界面

## 5.3 本章小结

本章主要阐述了隐写和隐写分析演示系统的设计和操作流程。首先介绍了系统的登录和注册操作，之后介绍了系统如何对图片进行隐写嵌入，如何进行隐写分析。其中隐写嵌入用到了第2章中生成含密图像使用的可执行文件；隐写分析用到了第3章中训练好的9个模型参数文件，隐写分析时系统根据使用者在前端选择的嵌入率和隐写术确定使用哪一个模型参数文件实例化模型进而对图片进行隐写分析并给出判断结果。

结 论

本文主要实现了基于深度学习的图像隐写分析，实验结果证明对于WOW、HUGO和S-UNIWARD三种隐写术，SRNet网络模型均具有较好的检测性能。此外，通过在原始SRNet网络模型中增加CBAM注意力模块，模型可以更好的进行特征提取，改进后的网络模型其隐写分析性能相比原始SRNet模型有了进一步提高。之后本文利用自编码器以及生成对抗网络实现了图像隐写去除操作，在保证图像视觉质量的同时成功去除了含密图像中嵌入的隐秘信息；最后本文利用PyQT5创建可视化界面，实现了图像隐写及隐写分析演示系统。

本文的主要工作内容如下：

1. 搭建了深度学习环境；下载原始图片数据集及WOW、HUGO和S-UNIWARD三种隐写术代码；将原始图片缩放至规定大小，用三种隐写术，每种隐写术按0.4bpp、0.7bpp、1.0bpp三种嵌入率在原始图像内进行隐写嵌入，生成9种含密图像数据集。
2. 复现并改进SRNet网络模型。首先对SRNet隐写分析模型进行详细介绍，之后编码实现SRNet模型，在数据集上训练并测试模型的准确率。之后引入CBAM注意力模块，详细说明CBAM注意力模块中的通道注意力机制和空间注意力机制，以及如何将CBAM注意力模块添加到原始SRNet模型中。通过控制变量法，仅改变模型结构，证明添加注意力机制后模型的检测性能得到提升。
3. 利用自编码器以及GAN网络实现了图像隐写去除。向自编码器网络中输入含密图像，输出去除隐秘信息后的图像，保证在去除隐写信息的同时，图像的视觉质量不会受到太大影响。为进一步优化自编码器的性能，将其添加到GAN网络中进行对抗训练。
4. 实现了图像隐写及隐写分析演示系统，该系统可以演示如何对原始图像进行隐写嵌入以及如何对图片进行隐写分析。

本文实现的SRNet和DDSP模型可以较好的实现图像隐写分析以及隐写去除操作，但模型仍存在提升空间。例如利用SRNet模型进行图像隐写分析时，每当隐写术和嵌入率发生改变，都需要重新训练模型，这将耗费大量时间；在实现图像隐写去除时，自编码器生成的图像虽然从肉眼上来看具有较好的视觉效果，但其PSNR和MSE值不够优秀，需要进一步调整网络训练参数。

目前利用深度学习进行图像隐写分析依旧是主流。在未来，如何设计出更加优秀的隐写分析网络模型；如何将深度学习和传统隐写分析方法进行结合是图像隐写分析领域的研究趋势。此外，数字图像只是信息传递的载体之一，诸如视频、音频等其他信息载体的隐写嵌入和隐写分析也是未来重要的研究方向。

# 参考文献

1. Akhavan S, Akhaee M A, Sarreshtedari S. Images steganalysis using GARCH model for feature selection[J]. Signal Processing: Image Communication, 2015, 39: 75-83.
2. 付章杰, 王帆, 孙星明, 等. 基于深度学习的图像隐写方法研究[J]. 计算机学报, 2020, 43(9): 1656-1672.
3. Harmsen J J, Pearlman W A. Steganalysis of additive-noise modelable information hiding[C]//Security and Watermarking of Multimedia Contents V. International Society for Optics and Photonics, 2003, 5020: 131-142.
4. Lyu S, Farid H. Detecting hidden messages using higher-order statistics and support vector machines[C]//International Workshop on information hiding. Springer, Berlin, Heidelberg, 2002: 340-354.
5. Chen X, Wang Y, Tan T, et al. Blind image steganalysis based on statistical analysis of empirical matrix[C]//18th International Conference on Pattern Recognition (ICPR'06). IEEE, 2006, 3: 1107-1110.
6. Pevný T, Filler T, Bas P. Using high-dimensional image models to perform highly undetectable steganography[C]//International Workshop on Information Hiding. Springer, Berlin, Heidelberg, 2010: 161-177.
7. Holub V, Fridrich J. Designing steganographic distortion using directional filters[C]//2012 IEEE International workshop on information forensics and security (WIFS). IEEE, 2012: 234-239.
8. Holub V, Fridrich J, Denemark T. Universal distortion function for steganography in an arbitrary domain[J]. EURASIP Journal on Information Security, 2014, 2014(1): 1-13.
9. Li B, Wang M, Li X, et al. A strategy of clustering modification directions in spatial image steganography[J]. IEEE Transactions on Information Forensics and Security, 2015, 10(9): 1905-1917.
10. Fridrich J, Kodovský J, Holub V, et al. Breaking HUGO the process discovery[C]//International Workshop on Information Hiding. Springer, Berlin, Heidelberg, 2011: 85-101.
11. Holub V, Fridrich J. Random projections of residuals for digital image steganalysis[J]. IEEE Transactions on information forensics and security, 2013, 8(12): 1996-2006.
12. Shi Y Q, Sutthiwan P, Chen L. Textural features for steganalysis[C]// International workshop on information hiding. Springer, Berlin, Heidelberg, 2012: 63-77.
13. Tang W, Li H, Luo W, et al. Adaptive steganalysis against WOW embedding algorithm[C]//Proceedings of the 2nd ACM workshop on Information hiding and multimedia security. 2014: 91-96.
14. Denemark T, Sedighi V, Holub V, et al. Selection-channel-aware rich model for steganalysis of digital images[C]//2014 IEEE International Workshop on Information Forensics and Security (WIFS). IEEE, 2014: 48-53.
15. Tan S, Li B. Stacked convolutional auto-encoders for steganalysis of digital images[C]//Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific. IEEE, 2014: 1-4.
16. 翟黎明, 嘉炬, 任魏翔, 等. 深度学习在图像隐写术与隐写分析领域中的研究进展[J]. 信息安全学报, 2018, 3(6): 2-12.
17. Fridrich J, Kodovsky J. Rich models for steganalysis of digital images[J]. IEEE Transactions on Information Forensics and Security, 2012, 7(3): 868-882.
18. Qian Y, Dong J, Wang W, et al. Deep learning for steganalysis via convolutional neural networks[C]//Media Watermarking, Security, and Forensics 2015. International Society for Optics and Photonics, 2015, 9409: 94090J.
19. Xu G, Wu H Z, Shi Y Q. Structural design of convolutional neural networks for steganalysis[J]. IEEE Signal Processing Letters, 2016, 23(5): 708-712.
20. Bas P, Filler T, Pevný T. “Break our steganographic system”: the ins and outs of organizing BOSS[C]//International workshop on information hiding. Springer, Berlin, Heidelberg, 2011: 59-70.
21. Ye J, Ni J, Yi Y. Deep learning hierarchical representations for image steganalysis[J]. IEEE Transactions on Information Forensics and Security, 2017, 12(11): 2545-2557.
22. Boroumand M, Chen M, Fridrich J. Deep residual network for steganalysis of digital images[J]. IEEE Transactions on Information Forensics and Security, 2018, 14(5): 1181-1193.
23. Woo S, Park J, Lee J Y, et al. CBAM: Convolutional block attention module[C]//Proceedings of the European conference on computer vision (ECCV). 2018: 3-19.
24. Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative adversarial networks[J]. arXiv preprint arXiv:1406.2661, 2014.
25. Corley I, Lwowski J, Hoffman J. Destruction of Image Steganography using Generative Adversarial Networks[J]. arXiv preprint arXiv:1912.10070, 2019.
26. Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
27. Bengio Y, Simard P, Frasconi P. Learning long-term dependencies with gradient descent is difficult[J]. IEEE transactions on neural networks, 1994, 5(2): 157-166.
28. He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
29. Mao X, Li Q, Xie H, et al. Least squares generative adversarial networks[C]//Proceedings of the IEEE international conference on computer vision. 2017: 2794-2802.
30. Arjovsky M, Chintala S, Bottou L. Wasserstein generative adversarial networks[C]//International conference on machine learning. PMLR, 2017: 214-223.
31. Ledig C, Theis L, Huszár F, et al. Photo-realistic single image super-resolution using a generative adversarial network[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4681-4690.
32. Lin T Y, Maire M, Belongie S, et al. Microsoft coco: Common objects in context[C]//European conference on computer vision. Springer, Cham, 2014: 740-755.
33. Da K. A method for stochastic optimization[J]. arXiv preprint arXiv:1412.6980, 2014.
34. El Beji R, Saidi M, Hermassi H, et al. An Improved CNN Steganalysis Architecture Based on “Catalyst Kernels” and Transfer Learning[C]//International Conference on Digital Economy. Springer, Cham, 2018: 119-128.
35. Mnih V, Heess N, Graves A, et al. Recurrent models of visual attention[J]. arXiv preprint arXiv:1406.6247, 2014.
36. Hu J, Shen L, Sun G. Squeeze-and-excitation networks[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 7132-7141.
37. Hore A, Ziou D. Image quality metrics: PSNR vs. SSIM[C]//2010 20th international conference on pattern recognition. IEEE, 2010: 2366-2369.
38. Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity[J]. IEEE transactions on image processing, 2004, 13(4): 600-612.

# 攻读学士学位期间发表的论文和取得的科研成果

# 致 谢

时光荏苒，岁月匆匆。从江南到北国求学已近四载，自己的本科生涯也即将画上句号，在此我想对我的母校、老师、同学还有家人表示衷心的感谢。

首先感谢我的毕设教师吴艳霞老师、付岩老师以及自己未来的研究生导师刘邵辉老师，本论文的顺利完成离不开三位老师的悉心指导。从一开始的资料收集、到毕设开题、中期答辩、再到最后的论文写作和格式修改，三位老师都付出了大量的时间和精力，没有各位老师的耐心教导，就没有本次毕设的顺利完成。

感谢母校哈尔滨工程大学为我们提供的良好学习环境和条件，让我们每一天都可以不断的充实和提升自我。“大学至真、大工至善”的校训我将一生铭记于心，并在之后的人生际遇中将“至真”、“至善”作为自己人生的座右铭。感谢四年来教导过自己的各位老师和辅导员，谢谢你们在学习和生活上对我的关心和帮助，也衷心的祝愿各位老师和辅导员身体健康、工作顺心，家庭美满。

感谢我的父母和家人们，家永远是最温暖的港湾，感谢你们这么多年来默默的支持和爱护，你们是我人生的全部，也是我一直努力奋斗和砥砺前行的动力所在，我爱你们。

最后感谢我的同学和朋友们，四年同窗，这期间有太多难以忘怀的点点滴滴，有太多的哭和笑，这将是我宝贵的人生记忆，我将永远珍藏在心间。