



毕业设计（论文）

装
订
线

课题名称 基于深度学习的服装图像自动标注方法

学 院 软件学院

专 业 软件工程

学生姓名 李源峰

学 号 1852448

指导教师 刘琴

日 期 2022.5.31

基于深度学习的服装图像自动标注方法

摘要

随着电子商务以及智能时尚领域的发展，电商已经逐渐成为服装销售的主流渠道。在服装被放到网络上进行销售前，首先需要将服装信息录入系统。在传统服装的信息录入时，还会通过人工标注的方式为服装图像打上材质、类别、风格等多种标签。这些标签在服装检索、自动推荐、智能穿搭、时尚分析等多个领域都存在极大的应用价值。但是，随着业务规模的增加，服装信息越来越多、类别越来越丰富，手动标注服装图像存在的效率低、错误率高等局限性也逐渐凸显。与此同时，近年来，深度学习与计算机视觉领域飞速发展，尤其是用作图像处理任务的卷积神经网络(CNN)。卷积神经网络在各类图像分类任务中都获得了很高的准确率，并且逐渐演变出了一些经典的网络结构设计。所以，本文对如何使用深度学习技术来替代人工标注，实现服装图像自动标注进行了研究。本文基于主流的卷积网络技术和相关的先验工作设计了4种多任务卷积神经网络理论结构，同时对于每种网络结构设计了大量实验来全面分析模型参数设置，统计分析了相关实验数据，并且基于实验结果对每个模型进行了总结性的性能和应用场景评述。

本文的主要贡献包括：1.在 Resnet, Rep VGG, Fashion Net 的理论基础上设计了4种多任务网络卷积神经来完成服装图像任务。2.进行了大量性能试验，其中基于 Fashion Net 改进并优化的 **New Fashion Net** 可以在缩减后的 Deep Fashion 数据集取得的 **87.9%/92.8%** 的分类 Top3/Top5 准确率和 **49.6%/53.2%** 的标签预测 Top3/Top5 召回率。

关键词：深度学习，智能时尚，计算机视觉，多任务卷积神经网络，多标签分类，多类别分类，属性预测，图像标注；

装
订
线

Automatic Clothes Labeling Based on Deep Learning

ABSTRACT

With the development of e-commerce and the intelligent fashion, e-commerce has gradually become the mainstream channel for garment sales. Before clothing is put on the Internet for sale, it is first necessary to enter clothing information into the system. In the traditional clothing information entry, the clothing images are also manually labeled with a variety of tags such as material, category, and style. These tags have great application value in many fields such as clothing retrieval, automatic recommendation, intelligent dressing, and fashion analysis. However, as the scale of business increases and the clothing information becomes more and more abundant, the limitations of low efficiency and high error rate of manual labeling of clothing images are gradually highlighted. Meanwhile, in recent years, the field of deep learning and computer vision has developed rapidly, especially the convolutional neural network (CNN) used for image processing tasks. Convolutional neural networks have achieved high accuracy rates in various image classification tasks and have gradually evolved some classical network structure designs. Therefore, this paper investigates how to use deep learning techniques to replace manual annotation and achieve automatic annotation of clothing images. This paper designs four theoretical structures of multi-task convolutional neural networks based on the mainstream convolutional network techniques and related a priori work, and designs a large number of experiments for each network structure to comprehensively analyze the model parameter settings, statistically analyze the relevant experimental data, and summarize the performance and application scenarios of each model based on the experimental results.

The main contributions of this paper include: 1. four multi-task networks convolutional neural based on Resnet, Rep VGG, and Fashion Net are designed to accomplish the clothing image task. 2. a large number of performance experiments are conducted, in which New Fashion Net, which is based on Fashion Net, is improved and optimized to achieve 87.9%/92.5% on the reduced Deep Fashion dataset. achieved 87.9%/92.8% classification Top3/Top5 accuracy and 49.6%/53.2% label prediction Top3/Top5 recall.

Key words: Deep learning, intelligent fashion, computer vision, multi-task convolution neural network, multi-label classification, multi-category classification, attribute prediction, image annotation;

目 录

1 引言	1
1.1 研究背景与意义.....	1
1.2 国内外研究现状.....	1
1.2.1 标注技术发展历程.....	1
1.2.2 服装数据集发展历程.....	3
1.2.3 卷积神经网络发展历程.....	3
1.2.4 小结.....	4
1.3 主要内容和结构安排.....	4
1.3.1 主要研究内容.....	4
1.3.2 论文主要创新点.....	5
1.3.3 论文结构.....	6
2 服装图像自动标注问题建模与服装数据预处理.....	7
2.1 服装图像自动标注问题建模.....	7
2.2 数据集存储结构梳理.....	7
2.3 Deep Fashion 数据预处理.....	9
3 基于 ResNet-50 的服装图像自动标注模型概述.....	10
3.1 引言.....	10
3.2 残差学习.....	10
3.3 批标准化.....	11
3.4 基于 Resnet50 服装图像自动标注网络结构.....	12
3.5 损失函数设计.....	15
4 基于 Rep - VGG 的服装图像自动标注模型概述	17
4.1 引言.....	17
4.2 类 VGG 网络结构的优势	17
4.3 Rep VGG 中的残差块	17
4.4 Rep-VGG 中的结构重参数化技术	18
4.5 基于 Rep VGG A 的服装图像自动标注模型结构	19
4.6 损失函数设计	20
5 Fashion Net 服装图像自动标注模型概述	21
5.1 引言.....	21
5.2 Fashion Net 网络结构	21
5.2.1 基于 VGG-16 的全局特征提取部分	21
5.2.2 Landmark Prediction Branch 部分	22
5.2.3 Attention Branch 部分	23
5.3 Fashion Net 损失函数设计	25
6 New Fashion Net 服装图像自动标注模型概述	26
6.1 引言.....	26
6.2 基于 Resnet 的全局特征提取	26
6.3 基于 Rep_VGG 的边界值位置预测子网络	27
6.4 基于 Rep_VGG 的边界值位置预测子网络	28
7 实验与分析.....	31
7.1 引言.....	31
7.2 实验设备概述.....	31
7.3 评估方法.....	31
7.4 训练策略概述.....	32
7.5 实验设计思路.....	33
7.6 分类权重 W_c 与属性权重 W_s 实验	33

装
订
线

7.6.1	实验结果与分析.....	34
7.7	Cloth Resnet A 相关实验	35
7.7.1	迭代次数 epoch 实验	35
7.7.1.1	实验结果与分析.....	35
7.7.2	学习率 learning rate 和衰减速率 learning rate decay 实验	36
7.7.2.1	实验结果与分析.....	36
7.8	Cloth Resnet B 相关实验	38
7.8.1	学习率 learning rate 和衰减速率 learning rate decay 实验	38
7.8.1.1	实验结果与分析.....	39
7.9	Cloth Rep VGG 训练设置.....	40
7.9.1	训练迭代次数 epoch 实验	40
7.9.1.1	实验结果与分析.....	40
7.9.2	学习率 learning rate 和衰减速率 learning rate decay 实验	41
7.9.2.1	实验结果与分析.....	41
7.10	Landmark 预测权重 W1 实验.....	43
7.10.1	实验结果与分析.....	43
7.11	Fashion Net 相关实验	46
7.11.1	训练迭代次数 epoch 实验	46
7.11.1.1	实验结果与分析.....	46
7.11.2	学习率 learning rate 和衰减速率 learning rate decay 实验	46
7.11.2.1	实验结果与分析.....	47
7.12	New Fashion Net 相关实验.....	49
7.12.1	学习率 learning rate 和衰减速率 learning rate decay 实验	49
7.12.1.1	实验结果与分析.....	49
7.12.2	New Fashion Net 各部分优化效果实验.....	51
7.12.2.1	实验结果与分析.....	51
7.13	模型时效性能对比.....	53
8	结论和展望.....	55
8.1	结论.....	55
8.2	展望.....	56
8.2.1	数据增广.....	56
8.2.2	复杂化损失函数设计.....	56
8.2.3	精细化数据集.....	57
8.2.4	从自动标注到自动服装分割.....	57
8.2.5	从标注数据到智能时尚业务.....	57
参考文献	58
谢 辞	60

1 引言

1.1 研究背景与意义

近年来，时尚电子商业发展呈现井喷态势。根据中国服装协会发布的《2018-2019 年度中国服装行业发展报告》显示，2018 年到 2019 年中国网络服装销售总额同比增长了 22%，销售额达到 4452.8 亿元[1]。电商已经成为时尚消费的龙头。在商家提供网络时尚服务时，首先需要构建并维护一个服装信息数据库。该数据库中不仅仅需要包括服装的预览图和基本参数，还需要为这些服装打上多样的标签信息，即对服装进行标注。这些标签包括服装的属性，类别，材质和风格等。它们在服装检索、服装分类、自动推荐、智能穿搭、时尚分析等多个领域都存在极大的应用价值。目前，大部分的服装标注工作还是靠人工完成。随着服装数据库规模的增加，人工标注所存在的不足也逐渐成为了商家的一大困扰。首先，人工标注难免会出现纰漏，导致引入错误标签，降低用户体验；第二，服装信息数量十分巨大，若想完成所有服装的精细标注，需要耗费大量成本；第三，服装存在一些隐式信息，比如服装的风格，由于不同人对服装的理解是不同的，人为标注难免会造成信息差异。所以，服装图像自动标注技术代表着服装标注的未来，也成为了当下智能时尚领域的一大研究热点。

服装图像自动标注的难点在于，服装的种类非常丰富，图片包含的内容抽象复杂，底层的视觉特征和高层语义之间存在很大差异，这会导致计算机非常难提取图片中包含的隐式信息，即产生“语义鸿沟(Semantic Gap)”问题。近年来，快速发展的深度学习和计算机视觉技术为服装图像自动标注带来了技术革命。用于图像处理的卷积神经网络(CNN)已经在多种数据集上展现了惊人的分类准确率，例如 MINST, CIFAR, ImageNet 等。在处理不同的分类任务时，不同的研究团队也设计了丰富的网络结构。其中有些网络结构因为其极强的特征提取能力和创新的结构设置而成为经典，比如 AlexNet, VGG, Resnet 等。这些复杂的网络能够全面、高效地识别图像语义信息，并能够实现非常高的分类准确率。所以，本文将基于深度学习这一技术背景，研究并构建高准确率、低训练时效的服装图像自动标注模型。该系统能够较好地解决人工标注所存在的问题，提高服装信息标注的效率和准确率，对于智能时尚领域有重大意义。

1.2 国内外研究现状

1.2.1 标注技术发展历程

早期，服装的标注是通过构建与特征点相关的特征描述符结合非监督式分类器来实现的。特征点是图片上一些十分突出的不会因光照、尺度、旋转等因素而消失的点，比如角点、边缘点、暗区域的亮点以及亮区域的暗点，这些点中往往包含图片的一些关键信息。代表性的特征点提取算法有 SIFT (Scale-invariant feature transform)[2], HOG (Histogram of Oriented Gridients)[3] 等。而非监督式分类器则是一种早期的机器学习方法，它可以在训练数据中学习到一个确切的数学模型(比如线性回归问题中，它可以学到输入数据与输出数据的线性函数关系)，并依此模式推测新的实例。论文[4]在 2012 年提出了一种服装属性预测的方法。他们首先采用不同的特征提取方式提取服装不同类型的特征，比如 SIFT 特征和通过最大值滤波器(Maximum Response Filter)

提取的纹理特征，然后将这些特征描述子分别放入 SVM 分类器中进行分类。论文[5]则提出了一种基于 SIFT 提取并描述服装图像中一些明显特征的技术，这些特征包括人的性别、关节位置等。由于 SIFT 具有尺度不变性，该方法可以不受照片中人物姿势变化的影响。论文[6]提出了一个完整的服装预测 Pipeline。该 Pipeline 首先细化了特征提取部分，加入了一些更加些细粒度的服饰特征描述子，然后采用随机森林分类器替换了 SVM 分类器，并取得了更优秀的分类准确率。同时，为了免除人为提取服装区域的工作，也有许多研究尝试如何自动识别照片中服装区域，称为服装分割(cloth segmentation)。在[7]中，提出了一种基于 CRF(conditional random field algorithm) 概率图模型的服装分割算法。在[8]中，介绍了一种将人体姿势估计技术和基于 MRF(Markov random field)马尔可夫随机场算法结合起来的服装分割算法。该算法不仅可以较好地提取服装边界，其进行服装分割的时候产生的特征提取符还可以放入 SVM，帮助后续的分类任务。

尽管基于特征提取和监督式分类器的研究取得了可观的性能，但是在不少有关论文[9-11]中都提到了该方法存在的弊端：(1). 服装信息存在的特征非常多元，不同的服装信息与不同的标签所对应的数学模型存在很大差异。所以，如果标注不同的标签，需要分别训练多个分类器。这会使得整个过程非常繁琐；(2). 不同的分类器需要接受不同类型的特征描述符，需要构造许多特征处理方式。(3). 特征提取时主要获取的是服装的高级语义特征，但是低级语义特征(如像素级特征)往往被忽略了。所以，随着深度学习和卷积神经网络(CNN)的出现，服装图像自动标注逐渐向以 CNN 为框架转变。论文[11]创新地提出了利用多任务 CNN 来解决服装的标注问题。该论文设置了若干个子网络，每个子网络包含相同的卷积块结构并对应一个类型的标注任务。这样，整个网络可以同时训练，并可以对比 CNN 在不同标注任务中的性能差异。在[12]中，作者提出相似服装图像检索任务不应该直接处理图像，而是应该首先提取服装图像的高级语义特征，即属性标签，然后再用这种标签进行服装检索。该论文利用 NIN[13]这一复合型卷积神经网络来预测服装的属性标签。NIN 网络为了更好地提取感受野特征，用多层感知机(MLP)替代了传统的卷积操作。他们的实验证明，NIN 网络能很好地预测服装的高级语义特征。他们的研究成果将服装特征提取从传统的特征算子方法转变成了用 CNN 提取人能够理解的语义特征。他们还同时证明了，语义标签比 SIFT 描述符更加有助于服装检索。

2016 年，Ziwei Liu 团队提出了一个大型的数据集 Deep Fashion[14]。该数据集为卷积神经网络提供了充分的训练资源。该数据集的提出带动了基于卷积神经网络的服装图像自动标注技术的高速发展。在[14]提出的同时，Ziwei Liu 团队提出了在此数据集基础上工作的，一个能够兼具服装信息全局特征与局部特征的多任务预测网络 Fashion Net。该网络以 VGG-16 网络作为基础框架，首先通过 VGG-16 提取网络的全局特征。然后利用 Deep Fashion 数据集中人为标注的 landmarks 服装边界点为基准，提取服装的局部特征。最后，链接两部分特征，进入分类层进行预测。[14]还设计了一套标准的评估方法来验证模型性能。在该标准的评估下，Fashion Net 取得了大大优于[11]和[12]的网络性能。2017 年，论文[36]提出了运用能够动态调节自适应的 Dense Net 在 Deep Fashion 上完成任务，并取得了比 Fashion Net 更好的分类准确性能。2018 年，论文[37]在处理时装图像将设计了一套特征提取方案服装图像抽象为“时尚语法”，并设计了一种基于双向卷积递归神经网络(BCRNNs)的来处理服装的全局特征以及语法拓扑上的信息。该模型取得了超过 90% 的分类预测准确率和 50% 以上的服装属性预测准确率。目前，该模型的性能已经成

为了服装标注卷积神经网络的性能基线。

同时，基于 CNN 的目标检测算法 Region-CNN(RCNN)[15]也开始被应用在服装分割领域。RCNN 采用卷积神经网络替代 SIFT 等特征提取算子提取图像中每个可能的区域，然后将这些特征向量输入 SVM 分类器中，完成目标检测。论文[16]中提出了利用 Mask-RCNN 高效提取服装区域的方法，并以此为基础构造了 Deep Fashion2 数据集。该数据集再 Deep Fashion 数据集上再次完成了数据增广，并通过 RCNN 自动检测减少了人为标注边界值(Bouding Box)出现的失误。

1.2.2 服装数据集发展历程

随着服装分类、检索等智能任务的发展，相关的公开数据集也逐渐增多。图表 1 中展示了近几年来服装数据集的规模对比。

	DCSA[4]	ACWS[18]	WTBI[19]	DDAN[20]	DARN[21]	DEEP FASHION[14]
#图片数量	1856	145,718	78,958	341,021	182,780	>800,000
#类别+属性	26	15	11	67	179	1,050
#配对数(用于服装检索)	None	None	39,479	None	91,390	>300,00
#特征区域	None	None	Bounding Box	None	None	Bounding Box + landmarks

表 1.1 服装数据集参数概览

数据集的迅速发展为训练自动标注卷积神经网络提供了数据基础。在 Deep Fashion 数据集中，已经囊括了超过 800,000 张图片。其中，用于属性和类别预测的图片多达 289,222 张。并且，Deep Fashion 数据集不仅手动标注了服装的 Bouding Box 区域，还为每张图片提供了 4~8 个 landmarks 边界点标注。这些边界点使得与 Deep Fashion 数据集一同提出的网络结构 Fashion Net 能够兼顾服装图片的局部特征。

1.2.3 卷积神经网络发展历程

1998 年，Y. Lecun 首次针对手写字符的识别问题，首次提出了第一个卷积神经网络 LeNet[22]。该论文奠定了卷积神经网络的种种基本概念，比如卷积核设置、激活函数、反向传播等。但是，由于该网络并未取得监督式学习算法的性能，所以并未引发卷积神经网络的研究浪潮。在 2012 年，论文[23]中提出了 AlexNet，并在并在 2012 ILSVRC (ImageNet Large-Scale Visual Recognition Challenge) 中获得了第一名。该网络大大增强了 LeNet 的网络复杂程度，使用 GPU 来加速运算，提出了池化操作，并且对训练集进行了全面的数据增强。该网络优秀的性能引发了卷积神经网络的研究热潮。在 2014 年，Simoyan 和 Zisserman 在[24]中提出了 VGG 网络。该网络参照 NIN[13]的思想，采用小型卷积核多次卷积的方法，使得卷积神经网络达到了 20 层以上的深度。该网络以较小的运算开销在 2014 年的 Image Net 挑战上获得分类第二名，定位第一名。在 2015 年，Kaiming He 提出了 Resnet[25]。Resnet 通过引入残差块结构结合批优化(Batch

Normalization)技术，完美地解决了高层神经网络的出现的退化问题，使得神经网络可以达到上百层的结构。该网络在同年的 ILSVRC 上获得了分别拿下分类、定位、检测、分割任务的第一名。目前，Resnet已经成为应用最为广泛的深层网络框架。在绝大多数深层网络变种中，都存在类似的残差结构。在 Resnet 提出后，网络结构逐渐向深层化[17]、多分支[27]、特征处理复杂化[28]的方向发展，并根据不同的任务处理类型分化出了不同的特征提取设计。但是，也有研究，意识到更加深层的网络会大大增强训练的空间占用率和时间消耗。盲目地加深网络结构去换取细微的准确率提升是非常不经济的做法。所以，卷积神经网络的研究在近年来也出现了另一个方向，即不再增强网络的层数，而是寻求回归轻量级的网络结构，通过优化网络结构中的运算细节来增强网络性能。2021 年 Ding Xiao Han 团队提出的一种 Rep VGG[26]网络架构即为这方面的代表。该网络重新采用类似 VGG 的单路网络结构，通过引入类残差块和结构重参数化技术以及训练/验证分开部署的方法，使得其在能够逼近 Resnet 技术的同时，能够比 Resnet 训练速度快近一倍。

1.2.4 小结

从上述文献综述可见，高速发展的卷积神经网络已经为服装图像自动标注技术提供了有力的技术背景支撑，而完备的服装数据集为服装图像自动标注网络提供了优越的训练资源。卷积神经网络已经可以替代传统的特征提取和监督式分类器的方法，为服装图像自动标注带来新的革命。基于综述，本文将采用大规模的 Deep Fashion 这一大规模数据集作为训练数据集，对服装图像自动标注工作进行多任务机器学习建模，开展相关实验并采用[14]中的评估标准评估模型的性能。在模型选择方面，本文选择最具有代表性的深层网络 Resnet 和轻量级网络 Rep VGG。并且，本文还尝试将 Resnet 和 Rep VGG 的有关技术融合进 Fashion Net 当中，使得 Fashion Net 能够达到 SOTA 性能，并将优化后的 New Fashion Net 与目前服装图像自动标注领域的基线性能[37]进行数据对比与分析。

1.3 主要内容和结构安排

1.3.1 主要研究内容

本文的整体研究方案示意图如下：

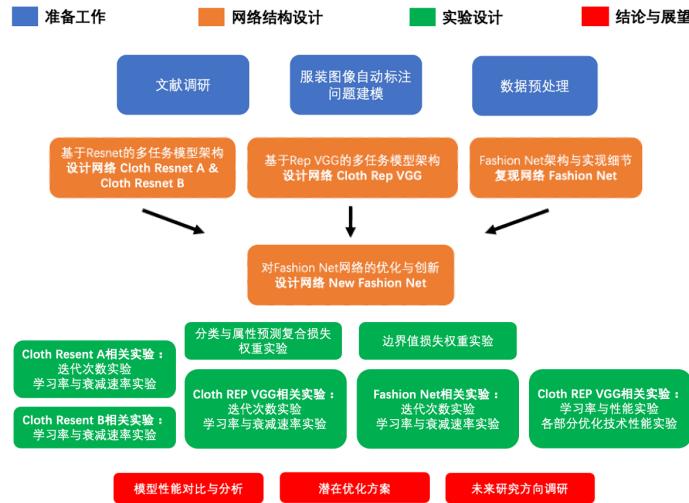


图 1.1 本文研究方案示意图

装
订
线

本文的研究内容可以划分为如下四个部分，具体内容如下：

- (1) 准备工作部分：在准备工作部分，本文首先对服装图像自动标注技术发展历史以及卷积神经网络的发展历史进行了全面的调研，然后对服装图像自动标注问题进行了数学建模，并完成 Deep Fashion 数据集的预处理工作，构建了本文所使用的数据集。
- (2) 网络结构设计部分：在网络结构理论部分中，本文首先研究了只能提取全局特征的多任务卷积神经网络。本文采用代表性的深层网络 Resnet50 和代表性的轻量级网络 Rep VGG A0 作为网络基础框架，在此基础上分别设计出了 **Cloth Resnet A**, **Cloth Resnet B** 和 **Cloth Rep VGG** 三种多任务网络。接着，为了利用 Deep Fashion 数据集中的边界值数据，增强模型的特征提取能力，本文对能够兼具局部特征与全局特征的 **Fashion Net** 开展了研究，详细调研了其模型结构与技术细节。最后，本文利用 Resnet50 和 Rep VGG 的技术原理，对 Fashion Net 的模型结构上进行了优化和创新，设计了 **New Fashion Net**。具体优化技术包括：将 Fashion Net 的全局特征提取部分由 VGG-16 框架替换为 Resnet50 框架，将边界点预测子网络的卷积核替换为 Rep Block，以及利用 SSIM 损失[29]重新设计 Fashion Net 的边界值损失函数。
- (3) 实验与分析部分：完成网络结构设计后，本文首先对多任务神经网络中的复合损失权重展开了实验，确认了应该使用的权重参数。然后针对五个网络分别设计了实验。实验确定了每个网络训练时的超参数设置，获得了每个网络的最佳性能。
- (4) 结论与展望部分：在获得每个网络的最佳性能后，本文对五个网络的性能进行了汇总和横向数据对比，同时将网络数据与基线数据进行了对比，总结了本文研究的结论，验证了 New Fashion Net 的性能优越性。之后，本文还调研了 New Fashion Net 的一些潜在优化方案和相关应用场景，提供了一些进一步研究此课题的思路。

1.3.2 论文主要创新点

本文主要创新点总结如下：

- (1) 将 Resnet50 单任务网络结构改造为了多任务网络结构用于完成服装图像自动标注任务。
- (2) 将 Rep VGG 单任务网络结构改造为了多任务网络结构用于完成服装图像自动标注任务。
- (3) 利用 Resnet 中的残差结构和 Rep VGG 中的 Rep Block 结构修改了 Fashion Net 网络结构，并利用 SSIM 损失重新设计了 Fashion Net 的边界值损失。优化得到的 New Fashion Net 可以在缩减到 200,000 张图片的 Deep Fashion 数据集上达到逼近基线数据[37]的性能，并且能在布料(Fabric)这一属性大类中取得比基线数据领先约 3% 的 Top3 召回率和约 4% 的 Top5 召回率。

1.3.3 论文结构

本文共 8 个章节，具体内容简要概括如下：

第一章：引言。本章简要分析了本研究的意义，对目前本文章研究领域的发展现状进行了全面的文献调研，介绍了本文的研究内容、思路与创新点。

第二章：自动标注问题建模和 Deep Fashion 数据集处理。本章首先对服装图像自动标注问题进行详细介绍了 Deep Fashion 数据集概况和存储结构，并介绍了如何将开源数据集转化为训练用数据集。

第三章：基于 Resnet50 的服装图像自动标注模型概述。本章介绍了如何利用现在业界最广泛的 Resnet 来构建一个深层的、性能较为优秀的多任务预测网络，并采用不同的分类链接方式构建了 Cloth Resnet A 和 Cloth Resnet B。详细介绍了两种网络的数学原理、网络架构、每层参数设置、损失设计和评估方法设计。

第四章：基于 Rep-VGG 的服装图像自动标注模型概述。本章介绍了如何利用新颖的 Rep – VGG 架构来构建一个轻便的、训练时效低的、性能较为优秀的多任务网络 Cloth Resnet VGG。详细介绍了该网络的数学原理、网络架构、每层参数设置。

第五章：Fashion Net 服装图像自动标注模型概述。本章介绍了 Fashion Net 的技术细节，包括该网络的数学原理、landmark 分支网络策略、Attention 机制、每层参数设置、以及复合损失设计。

第六章：New Fashion Net 服装图像自动标注模型概述。本章介绍了如何基于第三章和第四章中的网络模块结构，改进了 Fashion Net 网络的全局特征提取架构和边界值预测子网络。同时介绍了采用 SSIM 替换 Fashion Net 边界值损失的方法，增强模型在边界值上拟合的精度和收敛速度。

第七章：实验与分析。本章设计了实验确定每个模型设计的最优超参数，并统计了每个模型的最优秀性能，同时统计了每个模型的参数量和训练时效。

第八章：结论与展望。本章总结了第七章中的实验结果，得出了文章的结论。并且，对文章的未来研究方向进行简要介绍。

2 服装图像自动标注问题建模与服装数据预处理

2.1 服装图像自动标注问题建模

Deep Fashion 数据集中，共包含 48 个类别和 1000 个属性标签。一张图像只能属于一个类别，但是可以拥有多个属性标签。对于服装图像的类别分类任务，可以采用深度学习的图像多类别分类模型进行建模。多类别分类模型的建模方式为：给定图像矩阵 \mathbf{I} ，目标是拟合非线性函数 F ，其输出为一个类别预测向量 \mathbf{C} ， \mathbf{C} 中包含模型对各个类别的预测概率。其数学表达式为：

$$F(\mathbf{I}) = \mathbf{C} = (c_1, c_2, \dots, c_n), \quad n = 48, \quad \sum_{i=1}^n c_i = 1 \quad (2.1)$$

对于服装的属性标签预测任务，可以采用深度学习的多标签分类模型进行建模。多标签分类模型的建模方式为：给定图像矩阵 \mathbf{I} ，目标是拟合非线性函数 F ，其输出为一个属性预测向量 \mathbf{A} ， \mathbf{A} 中包含模型对每个属性标签的预测概率。其数学表达式为：

$$F(\mathbf{I}) = \mathbf{A} = (a_1, a_2, \dots, a_n), \quad n = 1000, \quad 0 \leq a_i \leq 1 \quad (2.2)$$

由于服装图像的自动标注需要同时解决上述两个问题，所以可以采用深度学习中的多任务模型(Multi-Task Model)来对服装图像自动标注任务建模。所以，将服装图像自动标注问题建模如下：给定图像矩阵 \mathbf{I} ，目标是拟合非线性函数 G ，其同时输出类别预测向量 \mathbf{C} 和属性预测向量 \mathbf{A} 。其数学表达式为：

$$G(\mathbf{I}) = \mathbf{C}, \mathbf{A} \quad (2.3)$$

2.2 数据集存储结构梳理

本文所用的数据集自港中文大学提供的服装公开数据集 Deep Fashion[14]中的 Category and Attribute Benchmark 子集。经笔者统计，该子集的基本参数如下：

Image	Category	Attribute	Bounding Box	Landmarks
289, 222	48	1000	True	4~8

表 2.1 Category and Attribute Benchmark 子集参数概览

该数据集共含 29,222 张图片，共制定了 48 个类别，1000 个属性标签。其中，1000 个属性标签又被细分为 Texture, Fabric, Shape, Part, Style 五个标签大类。数据集预览如下所示：



图 2.1 Deep Fashion 数据集预览（引自[14]）

该数据集中，每个图片都有对应的服装数据区域 Bounding Box。Bounding Box 为两个坐标 $(x_1, y_1), (x_2, y_2)$ 组成的矩形区域。下图为用 Matplotlib 绘制的 Bounding Box 提取服装区域预览结果：



图 2.2 Bounding Box 提取服装边界预览

该数据集中，每张图片都人为的标注了 4~8 个边界点 landmarks。这些边界点包括左/右衣领、左/右袖口、左/右腰围和左/右下摆。这些边界点的领域像素可以较好地反应服装局部关键特征。边界点数据是包含 $(x, y, visibility)$ 的三元组。其中 visibility 代表该像素点是否有效。如果边界点被其他物体遮挡，则该边界点无效。下图为用 Matplotlib 绘制的 landmarks 预览结果：



图 2.3 landmarks 数据预览

装

订
线

2.3 Deep Fashion 数据预处理

鉴于本文实验设备的限制，本文在将原数据集缩减到 200,000 张图片。缩减时，首先采用 Excel 读取 Deep Fashion 的数据索引，然后采用随机函数打乱索引顺序，再截取 200,000 条作为处理条目。经过处理后，通过 Excel 统计得到最多的类别数为 Dress(6893)，最少的类别数为 Kimono(1287)，说明处理后数据集的每个类别数量级均分布在一个相似的区间内，数据缩减并不会产生样本分布不均匀的情况，以至于严重影响模型性能。

根据 Deep Fashion 数据集存储结构，本文还设计了两种预处理方式，分别对应不需要使用需要实用 Landmark 数据的网络。设计的数据集预处理如下：



图 2.4 本文 Deep Fashion 数据集预处理流程

对于不需要 landmark 的数据，本文首先使用 Bounding Box 值提取服装所在区域，然后使用双三次线性插值法将图像转换为 224*224 的统一大小。接着读取对应的标签和属性向量。对于需

要 landmark 的数据，在提取 Bounding Box 后还需要重新校准 landmark 的坐标。

本文按照 7:3 划分数据集的训练集与验证集。

3 基于 ResNet-50 的服装图像自动标注模型概述

3.1 引言

2015 年，Kaiming He 及其团队提出了 Resnet 深层神经网络[25]。在该网络提出前，虽然已有文献[27]证明，更深的卷积神经网络可以具有更好的特征提取能力，获得更好的分类性能，但是卷积神经网络的深度一直维持在 20 层~30 层左右。这是因为一旦卷积神经网络的层数加深，由于反向传播会放大误差，会产生梯度消失和梯度爆炸问题。并且，由于深层卷积神经网络的参数规模非常庞大，模型非常难以训练，还会存在深层网络的性能退化问题，即随着网络深度的加深，模型的性能反而会降低。Resnet 通过 Batch Normalization[27]和残差块(Residual Block)较好地解决了这个问题。论文实验证明了 Resnet 具有非常优秀的分类能力，并且可以达到非常深的深度来获取更好的性能。论文还提出了几种经典的 Resnet 的结构 Resnet18, Resnet50, Resnet101 等。目前，Resnet 已经成为应用最为广泛的卷积神经网络。基于 Resnet 的优秀性能，本文首先设计了一个基于 Resnet50 的服装图像自动标注模型。由于服装图像自动标注是多任务分类，所以本文在 Resnet50 的最后一个特征层进行了两种不同的调整。本章将详细介绍 Resnet 的网络结构和基于它的服装图像自动标注模型设计。

3.2 残差学习

Resnet的最大创新之处在于残差块的引入。该论文提出，深层的网络不应该存在退化问题，因为不论网络深度如何加深，始终存在下图所示的网络：

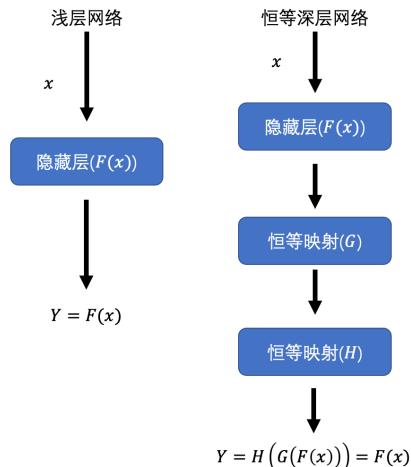


图 3.1 恒等映射网络

该网络通过不断在浅层网络中堆叠恒等映射层(identity mapping)来增加网络深度。这些恒等映射层不会影响网络的任何性能，所以，深层网络在最差的情况下，性能也应该是和浅层网络一致的。同时，作者认为，因为浅层网络已经取得了较优的拟合程度，所以深层网络的参数应该是浅层网络的恒等映射的近似。在这两个观点的基础上，作者设计了残差学习模块。假设数

据期望的基础映射为 $H(x)$ ，残差学习模块会让堆叠的非线性层去拟合残差映射 $F(x) = H(x) - x$ 。此时，基础映射便可以表示为 $H(x) = F(x) + x$ 。由于深层网络中，深层网络的基础映射 $H(x)$ 近似于 x 的恒等映射，那么 $F(x)$ 的值就会趋近于0。所以在实际训练时，很快就能找到使得模型收敛的 $F(x)$ 。残差学习模块的结构如图表8所示：

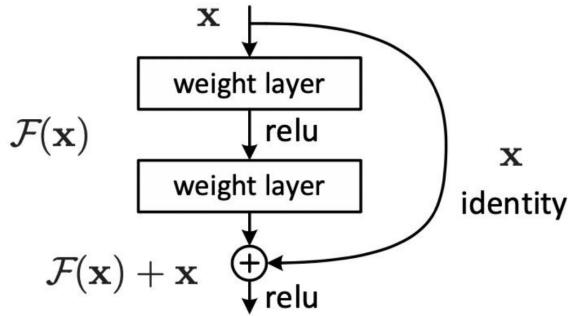


图 3.2 Resnet 的残差学习模块（引自[25]）

装
订
线

其数学表达式为：

$$y = F(x, \{W_i\}) + (H)x \quad (3.1)$$

其中， x 为上一层的输入， y 为到下一层的输出， F 为待学习的隐藏层变换， $\{W_i\}$ 为隐藏层权重。该变换要求 $F(x, \{W_i\})$ 与 x 的纬度一致。如果纬度不一致，则需要摄影变换 H 调整 x 的维度。

该公式同时表明，残差模块并没有增加额外的参数和计算复杂度。所以，引入残差模块并不会增加网络的训练难度。论文的实验证明，具有残差模块的网络比传统的卷积神经网络更容易收敛。论文中提到的 Resnet18, Resnet50, Resnet101 均由该残差块复合而成(残差块中的隐藏层和链接方式略有差异)。

3.3 批标准化

批标准化(Batch Normalization)最早在 Google 团队于 2015 年提出[25]。在 Resnet 中，该方法得到了非常广泛的应用。在深层的神经网络中，由于复杂的反向传播机制，如果输入存在极端值，很容易产生梯度爆炸或者梯度消失问题。Batch Normalization 通过将输入向量 x 归一化区间来解决这个问题。其数学表达式如下所示：

$$x' = \frac{x_i - \text{mean}(x)}{\sqrt{\text{std}(x)^2 + \epsilon}} \quad (3.2)$$

其中， $\text{mean}(x)$ 为均值， $\text{std}(x)$ 为标准差， ϵ 为偏置。在完成归一化后，还需要进行一次线性变换，其数学表达式如下所示：

$$y = \gamma x' + \beta \quad (3.3)$$

该线性变化用于调整归一化的程度。如果归一化过强以至于影响预测结果，神经网络在反向传播时，就会用该参数抑制归一化。

在 Resnet-50 中，每一个卷积层和激活函数层之间，都设置了批归一化操作。

3.4 基于 Resnet50 服装图像自动标注网络结构

由于实验设备限制，本文采用了 Resnet-50 作为网络的基本架构。由于服装图像自动标注任务分别包含类别标注和属性标注两部分任务，所以该任务是一个多任务分类问题。在实际设计时，网络首先采用 ResNet50 的前 48 层(除去平均池化和全链接层)作为特征提取网络。该部分的网络参数如下：

Layer name	Param [(n)代表次数]	Output Shape (Input: [batch_size,3,224, 224])
Conv_Block_1	Conv(3*3,64, stride = 1, padding =1)	[batch_size, 64, 112,112]
Batch Normalization	Batch Normalization	[batch_size, 64, 112,112]
Max_Pooling_Layer	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 64, 56,56]
Residual_Block_1*3	Conv(1*1, 64 ,stride=1, padding =0) Batch Normalization Conv(3*3, 64. Stride = 1, padding = 1) Batch Normalization Conv(1*1, 256, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	[batch_size, 256, 56, 56]
Down_Sampling_Layer	Down_Sample_Conv(1*1,256,Stride=2, padding=0)	[batch_size, 256, 28, 28]
Residual_Block_2*4	Conv(1*1, 128,stride=1, padding =0) Batch Normalization Conv(3*3, 128, Stride = 1, padding = 1) Batch Normalization Conv(1*1, 512, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	[batch_size, 512, 28, 28]
Down_Sampling_Layer	Down_Sample_Conv(1*1,512,Stride=2, padding=0)	[batch_size, 512, 14, 14]
Residual_Block_3*6	Conv(1*1, 256,stride=1, padding =0) Batch Normalization Conv(3*3, 256, Stride = 1, padding = 1)	[batch_size,1024, 14, 14]

装
订
线

	Batch Normalization Conv(1*1, 1024, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	
Down_Sampling_Layer	Down_Sample_Conv(1*1,512,Stride=2, padding=0)	[batch_size, 1024, 7, 7]
Residual_Block_4*3	Conv(1*1, 512,stride=1, padding =0) Batch Normalization Conv(3*3, 512, Stride = 1, padding = 1) Batch Normalization Conv(1*1, 2048, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	[batch_size,2048, 7, 7]

表 3.1 Resnet 50 特征提取部分网络参数

在最后两层的设计中，本文采用了两种结构设计，以下分别称为 Cloth Resnet A 和 Cloth Resnet B。在 Cloth Resnet A 中，执行属性预测的子网络与执行类别预测的子网络将共享特征提取层中的特征。该设计方案假设之前的已经可以提取服装图像中的所有关键特征，所以不需要对两种不同的分类任务加入额外的层细化处理特征图。所以，该网络的最后两层直接对特征图进行平均池化，然后将特征图进入全链接层，分别进行属性预测和分类预测。其连接层参数如下：

Layer name	Param [(n)代表次数]	Output Shape (Input: [batch_size,2048,7,7])
Avg_Pooling_Layer	AvgPooling(7*7)	[batch_size, 2048]
Attribute_Out	Linear(2048, 1000)	[batch_size, 1000]
Category_Out	Linear(2048, 48)	[batch_size, 48]

表 3.2 Cloth Resnet A 分类连接层具体参数设置

网络结构如下所示：

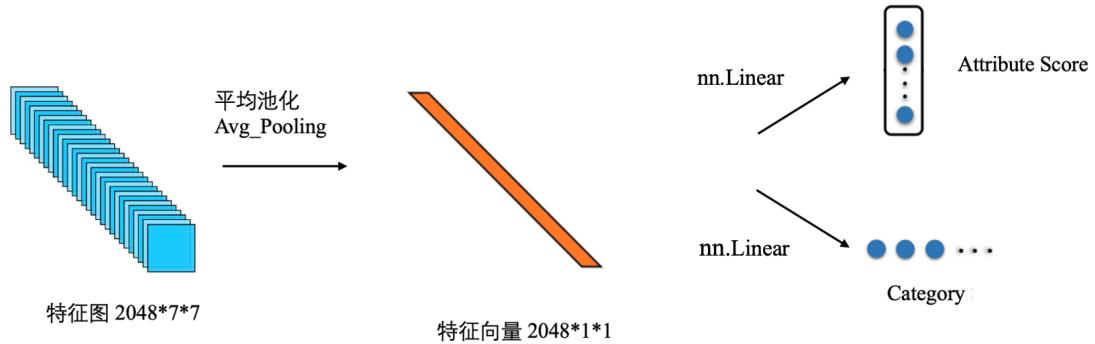


图 3.3 Cloth Resnet A 网络结构

在 Cloth Resnet B 中，网络的最后两层针对不同的分类任务进行了不同的设计。对于类别分类，由于 Category 更侧重于服装的全局特征，所以对其采用平均池化，然后进入两层全链接层的方式进行输出。增加了一层纬度为 1024 层的卷积层来增强特征提取能力。而对于属性分类而言，该问题是一个多标签预测问题。由于该问题涉及到更多的特征和属性标签，对于多标签预测问题，可能需要更深层的特征增强。由于 VGG-16 在 ImageNet 的千分类问题上取得了非常优秀的分类结果，所以 Cloth Resnet B 在最后层参照 VGG-16[24]的末层网络结构，重新设计了 3 层卷积层。其卷积核均参照 VGG-16 的小型卷积核。每次卷积操作后，都采用了批归一化和 Relu() 函数进行激活。其参数分别为：

Layer name	Param	Output Shape (Input: [batch_size, 2048, 7, 7])
Conv_1	Conv(3*3*512, stride = 1, padding = 1)	[batch_size, 512, 7, 7]
Conv_2	Conv(3*3*512, stride = 1, padding = 0)	[batch_size, 512, 7, 7]
Conv_3	Conv(3*3*2048, stride = 1, padding = 0)	[batch_size, 2048, 7, 7]
Avg Pool	AvgPooling(7*7)	[batch_size, 2048]
Linear	Linear(2048, 1000)	[batch_size, 1000]

表 3.3 Cloth Resnet B 分类链接层具体参数设置

Cloth Resnet B 的后两层网络结构如下所示：

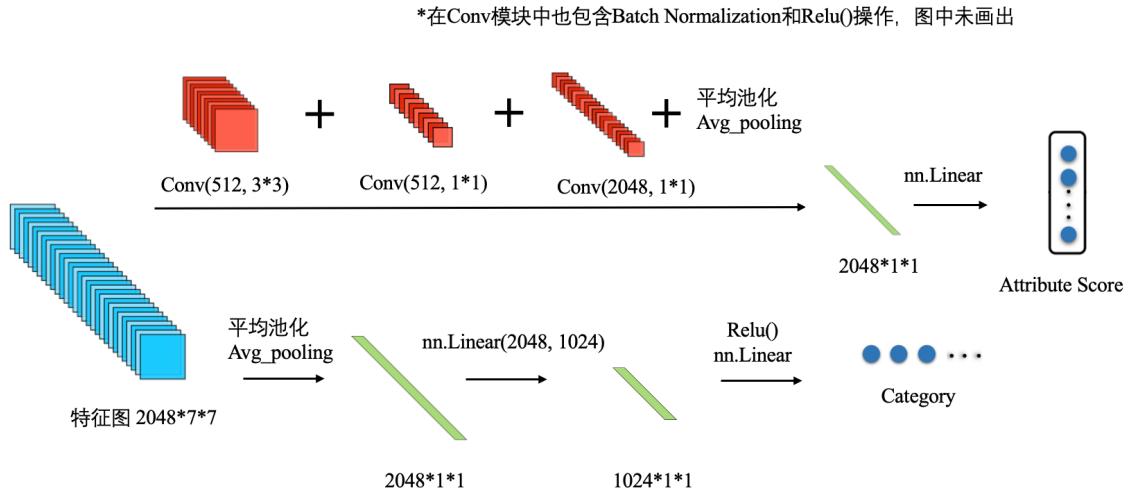


图 3.4 Cloth Resnet B 网络结构

实验章节将详细讨论两种网络的性能差异。

装

3.5 损失函数设计

由于本章需要将 Resnet 更改为多任务神经网络架构，包括自动标注分为类别 Category 和属性 Attribute 两样任务，所以本文需要重新设计网络的损失函数。

订

首先，对于类别损失函数，由于每种服装只包含一种类别，这是一个多类别分类问题。所以，本文采用多类别分类问题中应用最为广泛的多类别 Softmax+交叉熵损失函数。假设服装标注网络的输出为 l ，则对其进行 Softmax 变换可以得到其 Softmax 概率：

$$p_{i,j} = \frac{e^{l_{i,j}}}{\sum_{k=1}^C e^{l_{i,k}}} \quad (3.4)$$

线

其中， $p_{i,j}$ 表示第 i 个样本对第 j 类类别的预测概率。 $l_{i,j}$ 表示第 i 个样本对第 j 类类别的全连接层输出。Softmax 变换可以将 $l_{i,j}$ 固定为 $(0,1)$ 区间内的概率值。

而交叉熵损失的计算公式为：

$$L_{category} = \frac{1}{n} \sum_{i=1}^n \left(- \sum_{j=1}^C y_{i,j} \log(p_{i,j}) \right) \quad (3.5)$$

接着将概率代入交叉熵损失便可以得到 Category 损失的表达式：

$$L_{category} = \frac{1}{n} \sum_{i=1}^n \left(- \sum_{j=1}^C \log(p_{i,Y(i)}) \right) = \frac{1}{n} \sum_{i=1}^n \left(- \sum_{j=1}^C \log \left(\frac{e^{l_{i,Y(i)}}}{\sum_{k=1}^C e^{l_{i,k}}} \right) \right) \quad (3.6)$$

其中， n 为的 batch_size， C 为 Category 种类总数， $l_{i,Y(i)}$ 表示第 i 个样本在其所属真实类别 $Y(i)$ 上的输出值。

对于 Attribute 标注任务，这是一个多标签分类问题。对于多标签分类问题。多标签分类问

题和多标签分类问题的区别在于一个图像可能存在多个标签。所以，不能直接对该问题使用多类别分类的交叉熵函数。所以，本文将将该问题转化为一个 1000 维的二分类问题。然后使用带权重的 Sigmoid 激活+BCELoss 计算公式。该公式首先对输出结果采用 sigmoid 函数校准：

$$p(a_{ij}|x_j) = \frac{1}{1 + e^{-y_{ij}}} \quad (3.7)$$

其中，表示模型对第 i 张图片的第 j 类属性的预测概率，而 y_{ij} 表示模型对第 i 张图片的第 j 类属性的全链接层输出。接着，将上式代入多标签分类的 BCELoss 中，可以得到属性的损失函数表达式：

$$L_{attribute} = \text{mean} \left(\sum_{j=1}^n \sum_{i=1}^{1000} (w_{pos} \cdot a_{ij} \log(p(a_{ij}|x_j)) + w_{neg} \cdot (1 - a_{ij}) \log(1 - p(a_{ij}|x_j))) \right) \quad (3.8)$$

其中， n 代表网络的 batch_size， x_j 代表该批图像中第 j 张图像， a_{ij} 代表该第 j 张图像对应的属性标签， $p(a_{ij}|x_j)$ 代表网络对该标签的预测输出。 w_{pos} 和 w_{neg} 为假阴性和假阳性的惩罚系数，用于管理模型在正确属性和错误属性中的召回率(Recall)。由于在属性多标签预测时，我们更关注正确属性的召回率，所以在训练中，本文取 $w_{pos} = 1, w_{neg} = 0.1$ 。

在实际训练中，网络的总损失由 $L_{category}$ 和 $L_{attribute}$ 加权复合而成。其表达式为：

$$L_{all} = w_c \cdot L_{category} + w_s \cdot L_{attribute} \quad (3.9)$$

其中， w_c 和 w_s 用于权衡 $L_{category}$ 和 $L_{attribute}$ 的数量级差异，确保网络能够均衡地调整两个任务所涉及的参数。

装
订
线

4 基于 Rep - VGG 的服装图像自动标注模型概述

4.1 引言

2021 年, Ding Xiao Han 提出了一种 Rep VGG[26]的网络架构。该论文成功入选计算机视觉最顶会 CVPR 2021, 并引起了业界广泛关注。该网络架构的最大特点是结构简单但是性能优秀。这种网络架构非常简单, 它沿用非常经典的 VGG 网络结构, 即没有任何分支结构, 仅使用小型卷积核, 仅使用 Relu()作为激活函数。但是, 在维持简单网络架构的同时, Rep VGG 又通过引入残差块、和结构重参数化两种技术将其精度提高到了与 Resnet 非常接近的水平。鉴于 Rep VGG 的优秀性能, 并开展相关实验与分析。研究该网络的意义可以概括如下:

- (1) 研究一个能够快速完成训练且性能可观的服装图像自动标注模型。该模型在训练上的时效性具有很大的实用价值。
- (2) Rep-VGG 的模型结构与 VGG 非常相似。而与 Deep Fashion 一同提出的 Fashion Net 网络采用 VGG-16 作为基本框架, 且子网络中存在非常多的与 VGG 网络类似的卷积块。研究 Rep-VGG 可以为 Fashion Net 提供很好的的优化思路。

4.2 类 VGG 网络结构的优势

在卷积神经网络发展的过程中, 网络结构逐渐向深层化[17]、多分支[27]、特征处理复杂化[28]的方向发展。曾经取得非常优秀性能的 VGG 网络已经逐渐被人淘汰。但是, Rep-VGG 团队却在大量应用试验中发现 VGG 网络架构其实存在非常多的闪光点, 具体概括如下:

- (1) 在 VGG 网络架构中, 大量存在用 3×3 大小的卷积核进行卷积操作。在目前主流的深度学习框架当中(如 Pytorch, Tensorflow), 3×3 卷积运算都经过了专门的优化, 运算速度非常快。在 GPU 上, 3×3 卷积运算的计算密度(理论运算量除以所用时间)能达到 1×1 卷积运算和 5×5 卷积运算的 4~5 倍。
- (2) VGG 网络架构采用的是单路架构。在实现时, 单路架构的并行度高可以大大提高运行速度。同样的计算量, “大而整”的运算效率远超“小而碎”的运算。
- (3) VGG 网络的单路架构还可以非常好的节省内存。例如, ResNet 中的残差块结构虽然不会增加额外的运算量, 但是却增加了一倍的显存占用。
- (4) VGG 的单路架构具有很高的灵活性, 可以方便地修改卷积模块, 实现不同的移植或者优化工作。

综上所述, VGG 网络结构虽然非常简单, 但是具有很多复杂网络结构不具备的优势。所以, Rep-VGG 决定舍弃繁琐的网络设计, 重新研究 VGG 的网络结构, 让这个简洁而优雅的网络结构再次伟大。

4.3 Rep VGG 中的残差块

Rep VGG 能够获得高准确率的原因是因为 Rep VGG 在其结构中加入了类似 Resnet 的残差块结构。其残差结构如下图所示:

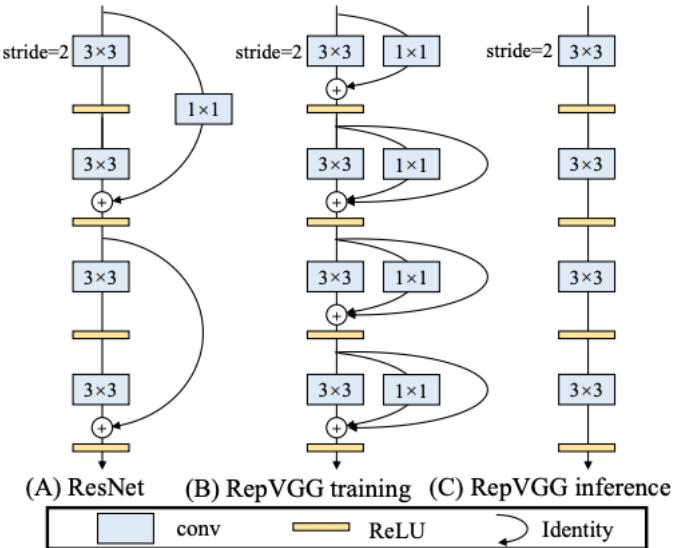


图 4.1 Rep-VGG 中的残差块设置(引自[26])

装

订
线

与 Resnet 不同的是，Resnet 中，残差块包含多个卷积结构。而 Rep-VGG 的残差块只包含一个 3×3 的卷积结构。并且，在 Resnet 中，采用的是“恒等映射”的链接方式，即残差块的输出直接与残差块之前特征层的输出相连。而在 Rep-VGG 中，采用的是一种复合链接的方式。其链接公式如下：

$$y = F(x, \{W_i\}) + G(x, \{W_i\}) + x' \quad (4.1)$$

其中 $F(x, \{W_i\})$ 代表一个 3×3 的卷积+批归一化激活操作。而 $G(x, \{W_i\})$ 代表一个 1×1 的卷积+批归一化操作， x' 代表 Rep-VGG 的残差块前层输出经过批归一化之后的结果。可以看到，Rep-VGG 新增加了一个 $G(x, \{W_i\})$ 链接因子，并且对每一个链接分子都进行了批归一化操作（Resnet 仅针对 $F(x, \{W_i\})$ 进行了批归一化操作）。Rep-VGG 经过这样设置之后，即可以获得像残差网络一样的易于训练的性能，又能通过改进的结构实现一种“结构重参数化”的技术，大大缩短模型的训练所用时间。

4.4 Rep-VGG 中的结构重参数化技术

Rep-VGG 虽然引用了残差块结构，但是，含残差块结构的模型是一个多路模型。多路模型会造成更大的时间开销。为了优化时效，使得 Rep-VGG 重新获得类似单路模型的性能，Rep-VGG 提出了一种“结构重参数化”的技术。该技术能够通过数学变换，将该残差块设置重新转换为一个 3×3 的卷积操作。该技术细节如下所示：

首先，Rep-VGG 会将三个链接因子全部转化为卷积操作。其中， 1×1 卷积操作等价于将其转化为以该值为中心，其余值均置为 0 的 3×3 卷积核。Identity 映射等价于一个 3×3 的，并且所有置均为一的 3×3 卷积核。

第二，由于卷积的每个链接因子含有批归一化操作。在重参数化时，需要将归一化操作融合到卷积里。转化公式如下：

$$W'_i = \frac{\gamma_i}{\sigma_i} W_i, b_i = -\frac{\mu_i \gamma_i}{\sigma_i} + \beta_i \quad (4.2)$$

其中, W_i 为第 i 个链接因子变化后的卷积核参数, b_i 为变化后的卷积运算偏置。 γ_i 和 β_i 为批归一化操作的尺度因子和偏置(见 3.2)。

接着, 由于卷积运算满足结合律, 可以将这三个卷积运算合并为一个卷积运算。其数学表达式如下:

$$y = x \circ W_1 + x \circ W_2 + x \circ W_3 = x \circ (W_1 + W_2 + W_3) = y \circ W \quad (4.3)$$

$$W = \sum_{i=1}^3 \frac{\gamma_i}{\sigma_i} W_i \quad (4.4)$$

$$b = \sum_{i=1}^3 -\frac{\mu_i \gamma_i}{\sigma_i} + \beta_i \quad (4.5)$$

其中, W 变化后的 $3*3$ 卷积核, b 为变化后的卷积运算偏置。结构重参数化技术可以可视化如下:

装
订
线

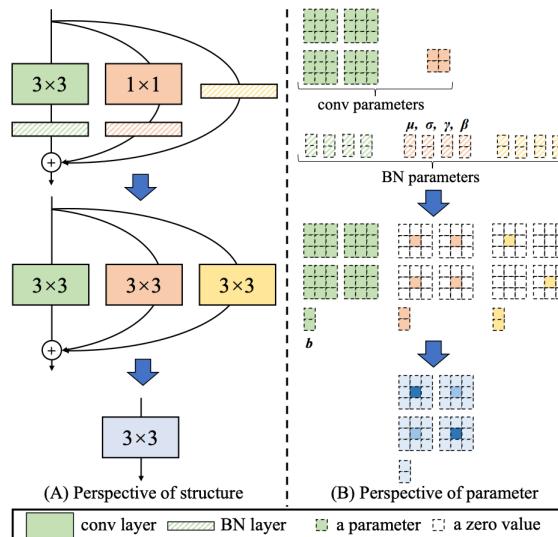


图 4.2 Rep-VGG 结构重参数化技术 (引自[26])

基于结构重参数化技术, Rep-VGG 提出了一种将训练与部署分离的训练技术。即 Rep-VGG 在训练时, 训练一个多分支网络。而在部署验证和测试时, 利用多分支技术将网络转化为单路网络, 进行验证。论文中的实验证明, 利用残差块+结构重参数化技术, 相似精度的 RepVGG 是 Resnet50 速度的 183%, Resnet101 速度的 201%。

4.5 基于 Rep VGG A 的服装图像自动标注模型结构

Rep – VGG 提出了两种网络结构, 分别为 Rep-VGG-A 和 Rep-VGG-B。其中, Rep-VGG-A 的参数较少, 可以获得近似 Resnet50 的性能。而 Rep-VGG-B 的参数较多, 可以获得类似于 Resnet101 的性能。鉴于实验设备的限制, 以及与 Resnet 模型组成对照性能对比, 本文基于 Rep-VGG-A 设计了 Cloth Rep VGG 网络结构。本文沿用了 Rep-VGG-A 的特征提取部分, 该部分共包含 21 个大小不同的 Rep-VGG 残差块, 然后参照 Cloth Resnet A (见 3.3 节) 的方式, 设计了到类

别的全链接层。该网络的参数如下所示(省略了每次 Rep_Block 和 Linear 后的 Relu() 激活)

Layer name	Param	Output Shape (Input: [batch_size, 3, 224, 224])
Rep_Block_1 *1	Conv3 (3*3, 48, stride = 1, padding = 1)	
	Conv1 (1*1, 48, stride = 1, padding = 0)	[batch_size, 48, 224, 224]
	Connection (Conv3_Out, Conv1_Out, Identity)	
Max_Pooling_Layer	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 48, 56, 56]
Rep_Block_2 *2	Conv3 (3*3, 48, stride = 1, padding = 1)	
	Conv1 (1*1, 48, stride = 1, padding = 0)	[batch_size, 48, 56, 56]
	Connection (Conv3_Out, Conv1_Out, Identity)	
Max_Pooling_Layer	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 48, 28, 28]
Rep_Block_3 *3	Conv3 (3*3, 96, stride = 1, padding = 1)	
	Conv1 (1*1, 96, stride = 1, padding = 0)	[batch_size, 96, 28, 28]
	Connection (Conv3_Out, Conv1_Out, Identity)	
Max_Pooling_Layer	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 96, 14, 14]
Rep_Block_4 *4	Conv3 (3*3, 192, stride = 1, padding = 1)	
	Conv1 (1*1, 192, stride = 1, padding = 0)	[batch_size, 192, 14, 14]
	Connection (Conv3_Out, Conv1_Out, Identity)	
Max_Pooling_Layer	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 192, 7, 7]
Rep_Block_5 *1	Conv3 (3*3, 1280, stride = 1, padding = 1)	
	Conv1 (1*1, 1280, stride = 1, padding = 0)	[batch_size, 1280, 7, 7]
	Connection (Conv3_Out, Conv1_Out, Identity)	
Avg_Pooling_Layer	AdaptiveAveragePool(7,7)	[batch_size, 1280, 1, 1]
Attribute_Output	Linear(1	[batch_size, 512, 7, 7]
Before_Linear	AvgPooling	
	DropOut	[batch_size, 4096]
	Linear(512*7*7, 4096)	
Attribute_Out	Linear(1280, 1000)	[batch_size, 1000]
Category_Out	Linear(1280, 48)	[batch_size, 48]

表 4.1 基于 Rep VGG A 的服装图像自动标注模型具体参数设置

4.6 损失函数设计

由于基于 Rep-VGG 的服装图像自动标注模型没有引入新的特征输入，模型输出与 Resnet 相同，所以仍采用 3.3 中的损失函数设计。

5 Fashion Net 服装图像自动标注模型概述

5.1 引言

提出 Deep Fashion 论文的团队在提出该数据集的同时，还提出了一种能够考虑服装局部信息的网络 Fashion Net[14]。在 Deep Fashion 数据集中，每张图片都增加了 4~8 人为标注的 landmark 边界点。这些边界点包括左右领口、左右袖口、左右衣宽和左右下摆。在 Fashion Net 中，设计了一个小型的子网络来提取该部分区域附近的特征，并将其结合到 VGG-16 提取的全局特征。该网络在类别分类任务中获得了突破性的 86% 的 Top-3 准确率，在属性多标签任务中也有非常优秀的表现。考虑到该网络的优秀性能，且该网络能够最大化 Deep Fashion 数据集的潜能，本文对该网络结构进行了细致的研究和代码复现，并开展了相关实验验证其性能。

5.2 Fashion Net 网络结构

Fashion Net 网络结构如下图所示：

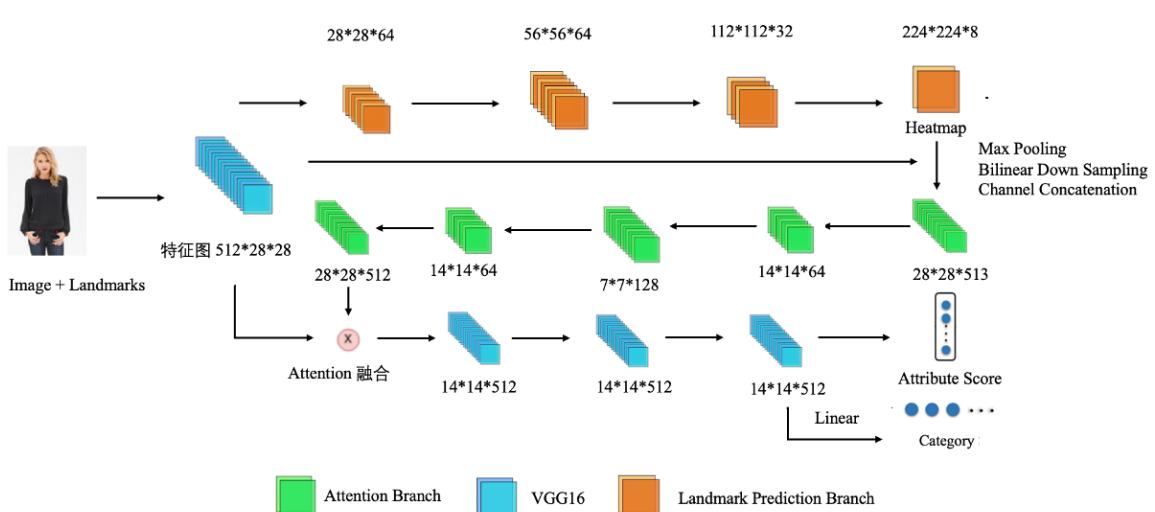


图 5.1 Fashion Net 网络结构

该网络具体可以细分为 3 个框架，分别为负责全局特征的 VGG-16，负责预测边界值点预测的子网络 Attention Branch，以及负责提取 landmark 位置的 Landmark Prediction Branch。

5.2.1 基于 VGG-16 的全局特征提取部分

VGG-16 网络由多个小型卷积核复合而成。多个复合的小型网络会提取图像的全局特征。Fashion Net 的截断了在 VGG-16 的第四个卷积块部分，提取出在这阶段中的特征图与 Attention Branch 输出的边界点对应特征进行特征融合，融合后继续进入剩余的 VGG 卷积块进行输出。对于最后的全链接层，Fashion Net 采用的类似 Cloth Resnet A 的全链接层结构，即没有对属性分类增加单独的卷积层。其具体参数如下所示(表格中省略了每次卷积后使用的 Relu() 激活函数)：

Layer name	Param	Output Shape (Input: [batch_size, 3, 224, 224])
------------	-------	--

Conv_1	Conv(7*7, 64, stride=2,padding=3)	[batch_size, 64, 112, 112]
Conv_Block_2	Conv(3*3,128, stride = 1, padding =1) (2) Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 128, 56, 56]
Conv_Block_3	Conv(3*3,256, stride = 1, padding =1) (2) Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 256, 28, 28]
Conv_Block_4(截断)	Conv(3*3,512, stride = 1, padding =1) (3)	[batch_size, 512, 28, 28]
特征融合	Output (Attention Branch)	[batch_size, 512, 28, 28]
Conv_Block_4	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 512, 14, 14]
Conv_Block_5	Conv(3*3,512, stride = 1, padding =1) (3) Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 512, 7, 7]
Before_Linear	DropOut Linear(512*7*7, 4096)	[batch_size, 4096]
Attribute_Out	Linear(4096, 1000)	[batch_size, 1000]
Category_Out	Linear(4096, 48)	[batch_size, 48]

表 5.1 Fashion Net VGG 全局特征部分参数设置

5.2.2 Landmark Prediction Branch 部分

在 FashionNet 中，专门设计了一个子网络，用于预测输入图片的 Landmark 所在位置。该网络首先采用 VGG-16 的连续小卷积核图像包含的 landmark 所属特征，然后采用逆卷积(Transposed Convolution)的上采样方法将特征图重新复原到与原图像一样大小，即 224*224 大小特征图。此时，特征图为包含原图像特征点的热图(HEAT MAP)。假设热图的值为 M' ，该热图参照的真实值应为以 landmark 真实值 x, y 为中心点，与图像大小相同的高斯核，如下图所示：

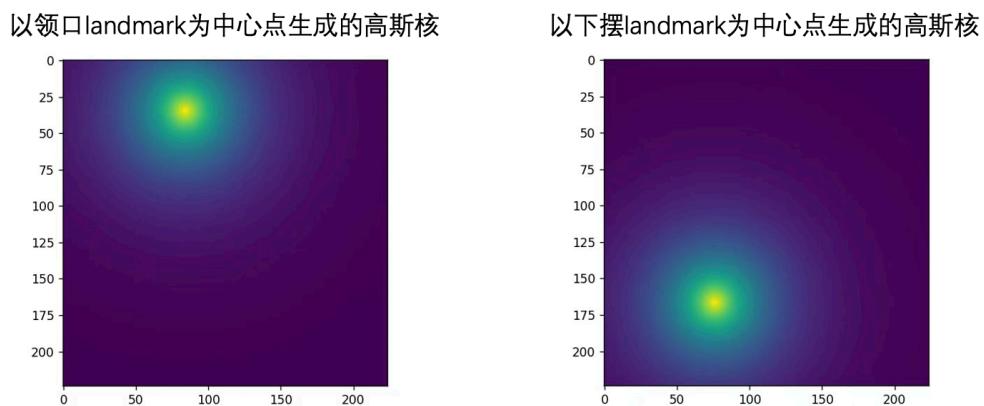


图 5.2 以 landmark 为中心生成的高斯核(以 matplotlib 绘制)

根据高斯核的性质，高斯核在核中心值的坐标将取得最大值。所以如果 M' 与 M 匹配成功，那么 M' 中最大值对应的下标就应该是正确的图像所含 landmark。评判 M' 与 M 的评估方法将在

Fashion Net 的损失函数设计中讨论。

Landmark Prediction Branch 具体参数如下所示(表格中省略了每次卷积后使用的 Relu()激活函数):

Layer name	Param	Output Shape (Input:[batch_size,512,28,28])
Conv_Block	Conv (1*1, 32, stride = 1, padding = 0) Conv (3*3, 64, stride=3, padding = 1) Conv (3*3, 64, stride=3, padding = 1) Conv (3*3, 128, stride=3, padding = 1) Transposed Conv (4*4, 64, stride = 2, padding = 1)	[batch_size, 64, 56, 56]
Conv_Block_2	Conv (3*3, 64, stride=3, padding = 1) Conv (3*3, 64, stride=3, padding = 1) Transposed Conv (4*4, 32, stride = 2, padding = 1)	[batch_size, 32, 112, 112]
Conv_Block_3	Conv (3*3, 32, stride=3, padding = 1) Conv (3*3, 32, stride=3, padding = 1) Transposed Conv (4*4, 16, stride = 2, padding = 1)	[batch_size, 16, 224, 224]
Conv_Block_4	Conv (3*3, 16, stride=3, padding = 1) Conv (3*3, 8, stride=3, padding = 1)	[batch_size, 8, 224, 224] (Heat map)

表 5.2 Fashion Net - Landmark Prediction Branch 部分具体参数设置

5.2.3 Attention Branch 部分

在完成输入图像的 landmark 预测后，生成的热图和 VGG 部分 Conv_Block_4 的输出会一并进入 Attention Branch。输入的热图维度为[batch, 8, 224, 224]。由于 8 张热图是类高斯核结构（只在 landmark location 处附近存在有效值，其他地方均为 0），所以生成的对输入在第二维度上求最大值，可以获得综合所有 landmark 信息的热图，其维度降为[batch, 224, 224]。接着，采用双线性采样的方式，将输入热图降维到[batch, 28, 28]。接着，热图和 Conv_Block_4 的输出链接，形成一个[513,28,28]的向量。该向量会经过类似 Landmark Prediction Branch 的卷积降维，逆卷积升维的过程。最终，形成一个[512,28,28]的向量。在卷积的同时，对该向量加入了 Batch Normalization 操作，并在最终使用了一个 Sigmoid 函数进行激活。该向量的所有参数将会被固定到(0,1)之间。

Attention Branch 具体参数设置如下所示(表格中省略了每次卷积/全链接后使用的 Relu()激活函数):

Layer name	Param	Output Shape (Input:[batch_size,512,28,28])
Conv_Block_1	Conv (1*1, 32, stride = 1, padding = 0)	
	Conv (4*4, 64 , stride=2, padding = 1)	[batch_size, 64, 14, 14]
	Batch Normalization	
Conv_Block_2	Conv (4*4, 128, stride=4, padding = 1) Batch Normalization	[batch_size, 128, 7, 7]
Trans_Conv_Block_1	Transposed Conv	
	(4*4, 64, stride = 2, padding = 1)	[batch_size, 64, 14, 14]
	Batch Normalization	
Trans_Conv_Block_2	Transposed Conv (1*1, 128, stride = 1, padding = 0)	
	Transposed Conv (1*1, 128, stride = 1, padding = 0)	[batch_size, 512, 28, 28]
	Transposed Conv (4*4, 512, stride = 2, padding = 1)	
	Batch Normalization	
	Sigmoid()	

表 5.3 Fashion Net - Attention Branch 具体参数设置

Attention Branch 的输出是一个注意力矩阵。使用 Envision 工具包对该某一图像对应的矩阵进行可视化，如下图所示：

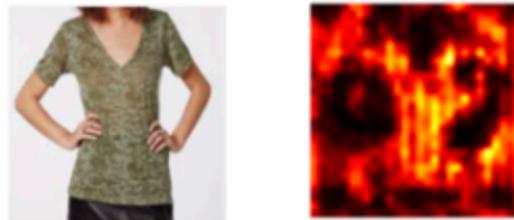


图 1 Attention Branch 输出可视化(通过 Envision 绘制)

该矩阵中，包含服装特征的坐标将具有较高值。接着，该矩阵会与 Conv_Block_4 的输出进入特征融合，融合公式为：

$$F_{attention} = (0.5 + A) \cdot F \quad (5.1)$$

其中， A 代表 Conv_Block_4 的输出， F 代表注意力矩阵， \cdot 代表矩阵对应元素相乘。0.5 的偏置项可以将 F 的值固定在(0.5, 1.5)之间，达到增强 landmark 对应特征弱化无关特征的效果。

5.3 Fashion Net 损失函数设计

Fashion Net 共包含三部分损失，分别是类别损失 $L_{category}$ ，属性标签损失 $L_{attribute}$ ，边界点损失 $L_{landmark}$ 。其中，类别损失和属性标签损失与之前网络使用的损失函数相同。对于边界点损失，Fashion Net 采用均值方差误差 MSE Loss 来衡量 Landmark Prediction Branch 输出热图和实际 landmark 对应高斯核之间的差异。其计算公式如下：

$$L_{landmark} = \text{mean} \left(\sum_{i=1}^{|D|} \sum_{j=1}^{L_i} \|M'_{ij} - M_{ij}\|_2^2 \right) \quad (5.2)$$

其中， $|D|$ 代表样本总数量， L_i 代表第 i 个样本所具有的 landmark 数量， M'_{ij} 代表网络对第 i 张图片的第 j 个 landmark 点输出的热图，而 M_{ij} 以代表第 i 个样本的第 j 个 landmark 点为中心点，以图片的长和宽为大小的高斯核。

将三个网络的损失函数加权求和，可以得到 Fashion Net 的损失函数。其表达式为：

$$L_{all} = w_c \cdot L_{category} + w_s \cdot L_{attribute} + w_L \cdot L_{landmark} \quad (5.3)$$

其中 w_L 是 landmark 是边界点损失的权重。在实际训练时， w_L 的取值应该较大，因为准确的 landmark 是 Fashion Net 运行的基础。要让 Fashion Net 的 Attention Branch 有效，必须首先保证 landmark 的预测值是准确的。所以， $L_{landmark}$ 必须快速完成收敛。

装
订
线

6 New Fashion Net 服装图像自动标注模型概述

6.1 引言

Fashion Net 的作者通过在数据集中引入 landmark 边界点，让神经网络能够增加服装局部地方的注意力，从而实现了网络性能的突破。这种加入局部网络的思路，是一种很好的优化网络方案。然而，FashionNet 对于全局特征和局部特征的提取均采用的是比较老旧的 VGG 结构，该结构的特征提取能力相较于近年的卷积神经网络是比较弱的。这说明 Fashion 在网络结构方面和还有很大的改进空间。在本章中，本文首先基于第三章和第四章理论基础，重新设计了 Fashion Net 的全局特征提取子网络结构和边界点预测子网络结构。同时，本文重新设计了 Fashion Net 对边界值的损失计算方式。这种新的损失方式可以加快模型的收敛速度。本章将对这三种优化技术进行简要介绍。

6.2 基于 Resnet 的全局特征提取

基于 Resnet 的研究结果，通过引入残差块，可以优化神经网络特征矩阵每次的精度，让神经网络更加精准地逼近合适值。本文通过第三章的相关实验(详见第七章)也发现，仅使用 Resnet，在不引入局部特征的情况下，在分类任务也能够达到超越 Fashion Net 的性能。所以，本文首先将 Resnet 的残差结构替换了 Fashion Net 的 VGG 全局特征提取网络部分。改进后的网络具体参数设置如下所示：

装
订
线

Layer name	Param (n)代表次数	Output Shape
	(Input: [batch_size,3,224, 224])	
Conv_Block_1	Conv(3*3,64, stride = 1, padding =1)	[batch_size, 64, 112,112]
Batch Normalization	Batch Normalization	[batch_size, 64, 112,112]
Max_Pooling_Layer	Max_Pooling(2*2, stride=2, padding=0)	[batch_size, 64, 56,56]
Residual_Block_1*3	Conv(1*1, 64 ,stride=1, padding =0) Batch Normalization Conv(3*3, 64. Stride = 1, padding = 1) Batch Normalization Conv(1*1, 256, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	[batch_size, 256, 56, 56]
Down_Sampling_Layer	Down_Sample_Conv(1*1,256,Stride =2, padding=0)	[batch_size, 256, 28, 28]
Residual_Block_2*4	Conv(1*1, 128,stride=1, padding =0) Batch Normalization Conv(3*3, 128, Stride = 1, padding = 1) Batch Normalization	[batch_size, 512, 28, 28]

装
订
线

	Conv(1*1, 512, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	
特征融合	Output (Attention Branch)	[batch_size, 512, 28, 28]
Down_Sampling_Layer	Down_Sample_Conv(1*1,512,Stride =2, padding=0)	[batch_size, 512, 14, 14]
Residual_Block_3*6	Conv(1*1, 256,stride=1, padding =0) Batch Normalization Conv(3*3, 256, Stride = 1, padding = 1) Batch Normalization Conv(1*1, 1024, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	[batch_size,1024, 14, 14]
Down_Sampling_Layer	Down_Sample_Conv(1*1,512,Stride =2, padding=0)	[batch_size, 1024, 7, 7]
Residual_Block_4*3	Conv(1*1, 512,stride=1, padding =0) Batch Normalization Conv(3*3, 512, Stride = 1, padding = 1) Batch Normalization Conv(1*1, 2048, Stride = 1, padding = 0) Batch Normalization Connection(Input, Conv_Output))	[batch_size,2048, 7, 7]
Avg_Pooling_Layer	AvgPooling(7*7)	[batch_size, 2048]
Attribute_Out	Linear(2048, 1000)	[batch_size, 1000]
Category_Out	Linear(2048, 48)	[batch_size, 48]

表 6.1 New Fashion Net 中的 Resnet 全局特征提取部分具体参数设置

6.3 基于 Rep_VGG 的边界值位置预测子网络

在 Fashion Net 的中，边界值位置预测子网络(即 Landmark Prediction Branch)采用的是一种类 VGG 单路架构。它由多个 3*3 的卷积块组成。这种类 VGG 结构非常适合用于实现 Rep-VGG 的残差优化。所以，本文尝试将 Fashion Net 的 Landmark Prediction Branch 的所有 3*3 卷积块替换为 Rep VGG 中的复合型残差卷积块，并在部署模型验证时将改残差卷积块利用结构重参数化技术重新转换为 3*3 卷积块，这样可以以仅增加少量训练时间开销，不增加验证时间开销为前提优化边界值预测子网络的精度。改进后的 Landmark Prediction Branch 具体参数设置如下(省略了每次 Rep_Block 和 Transposed Conv 操作后进行的 Relu()激活):

Layer name	Param	Output Shape (Input:[batch_size,512,28,28])
Conv_1	Conv (1*1, 32, stride = 1, padding = 0)	[batch_size, 64, 56, 56]
Rep_Block_1 *2	Conv3 (3*3,64, stride = 1, padding =1) Conv1 (1*1,64, stride = 1, padding = 0) Connection (Conv3_Out, Conv1_Out, Identity)	[batch_size, 64, 56, 56]
Rep_Block_2	Conv3 (3*3,128, stride = 1, padding =1) Conv1 (1*1,128, stride = 1, padding = 0) Connection (Conv3_Out, Conv1_Out, Identity)	[batch_size,128,56,56]
Transposed_Conv_1	Transposed Conv (4*4, 32, stride = 2, padding = 1)	[batch_size, 32, 112, 112]
Rep_Block_3 *2	Conv3 (3*3,64, stride = 1, padding =1) Conv1 (1*1,64, stride = 1, padding = 0) Connection (Conv3_Out, Conv1_Out, Identity)	[batch_size, 64, 112, 112]
Transposed_Conv_2	Transposed Conv (4*4, 16, stride = 2, padding = 1)	[batch_size, 16, 224, 224]
Rep_Block_4	Conv3 (3*3,8, stride = 1, padding =1) Conv1 (1*1,64, stride = 1, padding = 0) Connection (Conv3_Out, Conv1_Out, Identity)	[batch_size, 8, 224, 224]

表 6.2 加入 Rep Block 的 Landmark Prediction Branch 具体参数设置

6.4 基于 Rep_VGG 的边界值位置预测子网络

在 Fashion Net 中，需要评估 landmark 的预测位置与实际位置的差异。该网络的做法是，将 landmark 位置转换为一个与原图大小一样的高斯核，然后比较该高斯核与神经网络输出的热图的相似程度。在 FashionNet 中，采用了简单的 MSELoss 来解决这个问题(计算方法参见 5.3)。但是，MSELoss 存在的问题是，它无法评判两个二维矩阵在结构上的相似程度。以下面的情况为例：

真值	预测值1	预测值2																											
<table border="1"> <tr><td>0</td><td>-1</td><td>0</td></tr> <tr><td>-1</td><td>5</td><td>-1</td></tr> <tr><td>0</td><td>-1</td><td>0</td></tr> </table>	0	-1	0	-1	5	-1	0	-1	0	<table border="1"> <tr><td>0</td><td>-2</td><td>0</td></tr> <tr><td>-2</td><td>9</td><td>-2</td></tr> <tr><td>0</td><td>-2</td><td>0</td></tr> </table>	0	-2	0	-2	9	-2	0	-2	0	<table border="1"> <tr><td>2</td><td>0</td><td>2</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>2</td><td>0</td><td>2</td></tr> </table>	2	0	2	0	1	0	2	0	2
0	-1	0																											
-1	5	-1																											
0	-1	0																											
0	-2	0																											
-2	9	-2																											
0	-2	0																											
2	0	2																											
0	1	0																											
2	0	2																											

图 6.1 MSELoss 相同但结构不同的二维矩阵示例

图中预测值 1 和预测值 2 的 MSELoss 是相同的。但是，我们在实际预测中，肯定希望能朝着预测值 1 的方向优化。这是因为预测值 1 更加类似于一个高斯核的结构，虽然它不符合一个高斯核的具体值设置，但是我们肯定它能够强调其中心位置的信息，即边界值坐标所在点信息。

所以，MSELoss 是明显存在缺陷的。

为了解决这个问题，本文将 Fashion Net 用于评估边界值损失的函数替换为了图像结构损失 SSIM(structural similarity index) Loss[29]。该损失函数能够从两个矩阵的像素和、对比度、和结构三方面综合衡量两个矩阵的相似程度。该损失由三个方面复合而成，可以表示为：

$$SSIM(x, y) = f(l(x, y), c(x, y), s(x, y)) \quad (6.1)$$

假设现在需要比较的二维矩阵有 N 个元素，每个元素的值为 x_i ，那么其像素和的平均值为：

$$\mu_x = \frac{1}{N} \sum_{i=1}^N x_i \quad (6.2)$$

SSIM 用如下准则衡量两个矩阵像素和的相似度：

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \quad (6.3)$$

其中，常数 C_2 是为了防止 μ_x 或者 μ_y 为 0 的情况。在代码实现中，通常取值为 0.01。

对于两个矩阵的对比度，即是两个矩阵的值变化剧烈程度，该值体现在矩阵的标准差中，即：

$$\sigma_x = \sqrt{\left(\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)^2 \right)} \quad (6.4)$$

SSIM 采用相似的公式计算两个矩阵对比度的相似程度：

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \quad (6.5)$$

对于两个矩阵的结构，两个矩阵的结构应该由一个向量表示。同时，在研究两个矩阵的结构时，应该排除均值和标准差的数量级影响。所以，SSIM 衡量两个矩阵的结构差异时，首先需要将两个矩阵归一化为 $\mathbf{x}' = (\mathbf{x} - \mu_x)/\sigma_x$ ，然后计算两个向量的余弦相似度来获得 $s(x, y)$ ，其表达式如下：

$$s(x, y) = \frac{\mathbf{x} \cdot \mathbf{y} + C_3}{|\mathbf{x}| |\mathbf{y}| + C_3} = \frac{C_3}{(\sigma_x \sigma_y + C_3)} \left(\frac{1}{N-1} \sum_{i=1}^N (x_i - \mu_x)(y_i - \mu_y) \right) = \frac{\sigma_{xy} + C_3}{\sigma_x \sigma_y + C_3} \quad (6.6)$$

其中， σ_{xy} 表示 \mathbf{x} 和 \mathbf{y} 的协方差公式。在实际使用时，为了防止分母为 0，作者同样也加入了 C_3 其中，为了与 $c(x, y)$ 进行约分， C_3 取 $\frac{1}{2}C_2$ 。

将三个指标相乘便可以得到两个矩阵的 SSIM 差异指数：

$$SSIM(x, y) = \left(\frac{2\sigma_{xy} + C_2}{\sigma_x^2 + \sigma_y^2 + C_2} \right) \left(\frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1} \right) \quad (6.7)$$

由于在二维矩阵的维度上，均值和方差往往变化非常剧烈。所以，在实际应用时，SSIM 会采用一个正方形的滑窗以步长为 1 在图像上滑动，然后对于每个 patch 中的矩阵计算 SSIM 值，

然后取 SSIM 值的平均值作为 SSIM 的损失。

假如整张图有 M 个 patch，那么 SSIM 损失的计算公式为：

$$L_{SSIM} = \frac{1}{M} \sum_{j=1}^M SSIM(x_j, y_j) \quad (6.8)$$

本文将在 New Fashion Net 中采用 SSIM 损失 L_{SSIM} 函数替换原来的 $MSELoss$ 损失函数，以获得更好的热图拟合精度。而在[29]中，也介绍了 SSIM 损失因为是从三个维度调整像素值，在图像重建等领域时具有比 MSE 损失更好的收敛速度。所以，SSIM 损失还可能让 New Fashion Net 更快地完成收敛，从而减少训练迭代次数。

装
订
线

7 实验与分析

7.1 引言

本章将对理论部分中设计的 5 个网络结构设计多项实验并综合验证其性能。在实验设计时，首先对于设计的 5 个网络分别展开了实验，调整其超参数，以获得最优秀的网络性能。对于 New_Fashion_Net，本文将三个优化模块分别进行实验，探讨每个模块对 Fashion Net 的优化效果。在完成每个网络的实验与结果分析之后，将分别取训练效果最好的不同网络，结合服装图像自动标注参考文献中的经典网络，横向对照每个网络的性能。同时，实验还横向对比了不同网络的参数量和训练时间，该数据能反应不同网络的实现成本和训练的时效成本。

7.2 实验设备概述

本文的相关代码通过深度学习 Python 库 Pytorch 实现。Pytorch 是由 Facebook 人工智能研究院推出的机器学习工具包，该工具包对神经网络的一些基本参数进行了封装，包括模型类、反向传播、交叉熵损失。该工具还支持将模型迁移到 GPU 上，用 GPU 更强大的计算能力加速训练过程。

本文的实验数据图表绘制通过 TensorBoardX 和 Matplotlib 实现。Tensorboard 是 Google 推出的深度学习工具包 TensorFlow 的一个附加工具，可以方便地记录训练过程的数字、图像等内容，以方便研究人员观察神经网络训练过程。而 TensorBoardX 为 TensorBoard 在其他工具包中的移植版本，支持在 Pytorch 中进行使用。而 Matplotlib 是 Python 中的数据统计工具。实验将采用该工具绘制网络参数、训练时间等静态参数的对比图表。

本文的相关代码用 RTX 2080 GPU 进行训练。

7.3 评估方法

本文对于模型准确率的评估方法采用[14]中介绍的评估方法，此方法也是服装图像自动标注相关网络中通用的基线方法。对于类别，采用 Top-3 准确率，Top-n 准确率作为模型的衡量指标。其计算公式为：

$$Precision = \frac{\sum_{i=1}^{|D|} |y^{(i)} \cap y'^{(i)}|}{|D|} \quad (7.1)$$

其中 $y^{(i)}$ 表示第 i 个样本真值， $y'^{(i)}$ 表示第 i 个样本预测值， $|D|$ 样本数量。在 top-n 准确率中，只要预测结果的最高分值的前 n 项包含 $y^{(i)}$ ，则 $|y^{(i)} \cap y'^{(i)}| = 1$ 。

对于属性，采用 Top-n 召回率作为衡量指标。其表达式为：

$$Recall = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{\sum_{j=1}^{|C|} |y_j^{(i)} \cap y'_j^{(i)}|}{Y^{(i)}} \quad (7.2)$$

其中， $|D|$ 代表样本数量， $|C|$ 代表标签总数， $Y^{(i)}$ 代表第 i 个样本所拥有的标签总数，

$\sum_{j=1}^{|C|} |y^{(i)} \cap y'^{(i)}|$ 代表第 i 个样本预测正确的标签总数。在 top-n 准确率中，只要预测结果的最高分值的前 n 项包含 $y_j^{(i)}$ ，则 $|y_j^{(i)} \cap y'_j| = 1$ 。

7.4 训练策略概述

本文在训练神经网络时，采用批训练(batch_training)的方式对神经网络进行训练。在批训练中，每次会从数据集中拿出一批数据而不是全部数据输入网络，然后计算损失并通过反向传播优化参数。当所有数据都被拿进神经网络调整了一次参数后，神经网络完成了一次迭代(epoch)。在完成迭代后，神经网络会使用当前参数，对验证集进行准确率验证。接着，神经网络会继续进行下一次迭代，即再次将数据一批一批地放入神经网络中。当网络的损失趋于收敛时，神经网络的训练完成。这种训练方式可以大大增加训练时内存利用率，提高大矩阵乘法的并行化效率，同时降低训练时的显存占用率。考虑到实验算力，本文采用的训练批大小为 16，测试批大小为 32。由于接受一批数据，模型调整一次，所以每次接受批数据称为一步(step)。假设训练样本量为 D ，步数与训练迭代次数的关系为：

装
订
线

$$\frac{step * batch_size}{|D|} = epoch \quad (7.3)$$

本文采用预训练模型微调的训练方法。在 Pytorch 中，提供了许多训练好的模型。这些预训练模型都是在谷歌的开源数据集 ImageNet[32]上训练的。ImageNet 的数据量包括 14,197,122 张图片 1000 个类别。在此如此庞大且类别总数非常多的数据集上拟合的与训练模型，可以具有很好的对图像普适特征的提取能力，这往往比随机初始化模型参数更加有效，也能够更快达到收敛。

本文采用的优化算法为 Adam(Adaptive Momentum Estimation)[30]优化算法。Adam 优化算法是一种替代传统的随机梯度下降[31](SGD)的算法。传统的随机梯度下降算法会采用统一的学习率更新所有参数，但是在实际应用中，每个参数的重要性肯定是不一样的。这导致 SGD 很容易无法逼近最优解，而总是在最优解附近震荡。而 Adam 算法会累计每次梯度更新的一阶矩估计和二阶矩估计，通过这两个值将每个参数的变化值固定到一个合理的变化范围，增强模型的收敛能力。在论文中，论文[30]提到，建议使用 Adam 时，设置对应学习率为 0.001~0.0001 之间。

本文还采用一种学习率衰退(Learning Rate Decay)的策略。在神经网络训练初期，通过较大的学习率，快速找到到达临近收敛附近的值。而在收敛值附近时，采用较小的学习率进行试探，以防止在最优解附近震荡。训练时的实现方式为在每个 epoch 后将对当前学习率乘以衰减常数。下图为通过 TensorboardX 可视化的学习率随批数量变化情况：

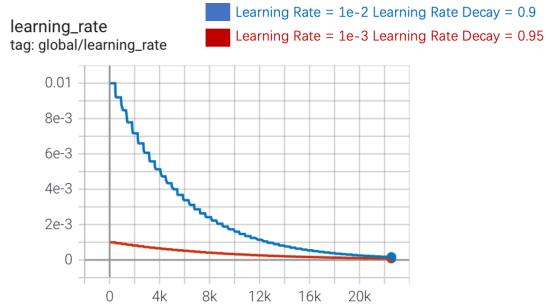


图 2.1 学习率衰减可视化

7.5 实验设计思路

本文的实验设计总思路为，首先开展确定多任务模型中各个损失的权重 W ，然后确定适合该网络的迭代次数 lim_epoch ，接着在该 epoch 下使用不同的学习率和衰减速率训练网络，获得网络的最佳性能。对于相似的网络结构，会省略迭代次数实验，直接进行学习率和衰减速率实验来获得网络的最佳性能。在 New Fashion Net 网络学习率和衰减速率实验完成后，本文还开设对 Fashion Net 网络的三种优化技术分别开展了实验，来探究每部分对网络结构的具体优化效果。实验的整体方案示意图如下：

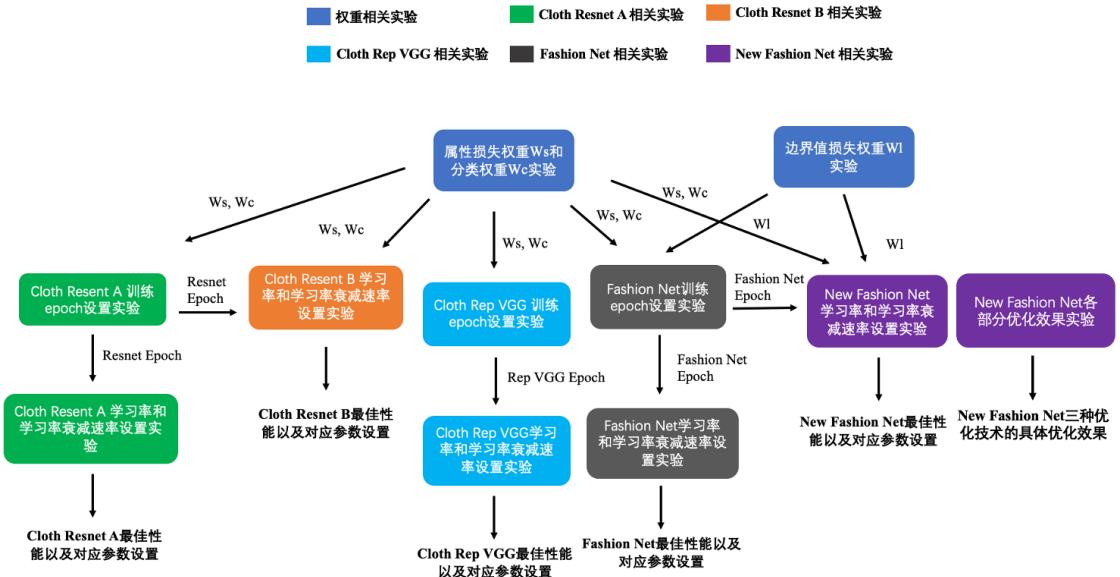


图 7.2 实验整体方案示意图

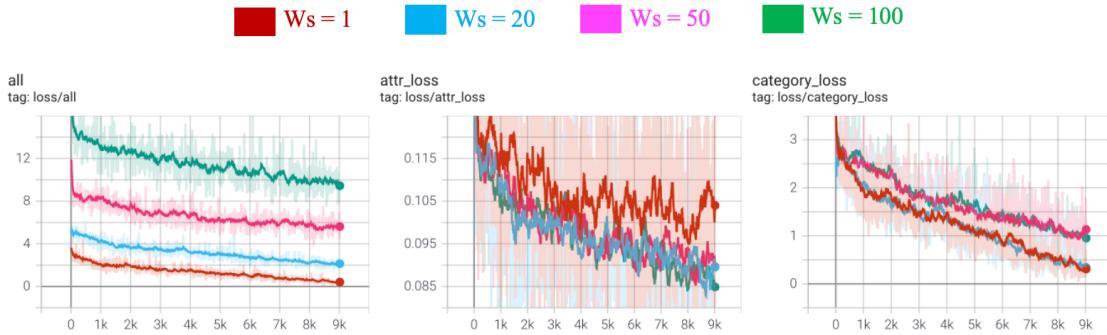
7.6 分类权重 W_c 与属性权重 W_s 实验

在多任务神经网络中，损失包含分类损失 $L_{category}$ 与 $L_{attribute}$ 的加权求和项。根据 3.4 的理论基础，分类损失 $L_{category}$ 应该与 $L_{attribute}$ 具有相近的数量级。这样可以保证神经网络均匀地反向传播，均衡地完成类别分类与属性分类任务。在实验前，首先根据代码计算得到最初的 $L_{category}$ 的数量级在(2,4)区间内，最初的 $L_{attribute}$ 数量级在(0.1, 0.3)区间内，可见，与理论相符的 W_s 应为 W_c 的 20 倍左右。

为了研究 W_c 和 W_s 的大小对网络性能的影响，在 Cloth Resnet A 上设置如下实验：首先固定 W_c 的值为 1，分别取 W_s 的值为 [1, 20, 50, 100]，均以 0.001 的 learning rate, 0.9 的 learning decay rate，训练 20 个 epoch，观察模型损失变化随步数变化情况和验证集性能随步数变化情况。

7.6.1 实验结果与分析

实验后，对应网络的 loss 如下图所示：



装
订
线

图 7.3 不同损失权重下的损失(加入了 0.5 的平滑)

根据该实验我们可以分析出当 W_c 固定为 1 时， W_s 的取值与模型性能的关系。当 W_s 取值过小时（图中红色曲线），分类损失 $L_{category}$ 可以收敛的很快。但是由于此时属性损失 $L_{attribute}$ 的数量级太小，导致 $L_{attribute}$ 非常难以收敛。而当 W_s 取值很大时（图中洋红色、绿色曲线），模型的属性损失 $L_{attribute}$ 可以较快地完成收敛，但是分类损失 $L_{category}$ 太小会收敛得很慢。并且，减小 W_s 并不意味着可以持续加速分类损失的收敛速度，因为图中当 W_s 取 20 与 W_s 取 1 时，分类损失的变化曲线是基本重合的。同理，增大 W_s 也不意味着可以持续加速属性损失的收敛速度，因为图中当 W_s 取 50 与 W_s 取 100 时，属性损失的收敛变化曲线是基本重合的。

综上所述，当 W_s 取 20 的时候，将可以获得最好的网络性能。因为此时分类损失和属性损失都可以良好的完成收敛。网络的准确率参数也印证了这个结论。下图分别为四种 W_s 取值对应的分类准确率和属性预测召回率的表格。可以看到， W_s 取 20 对应的蓝色曲线在属性预测方面能够取得非常突出的性能，在分类预测方面也有较为出色的性能。

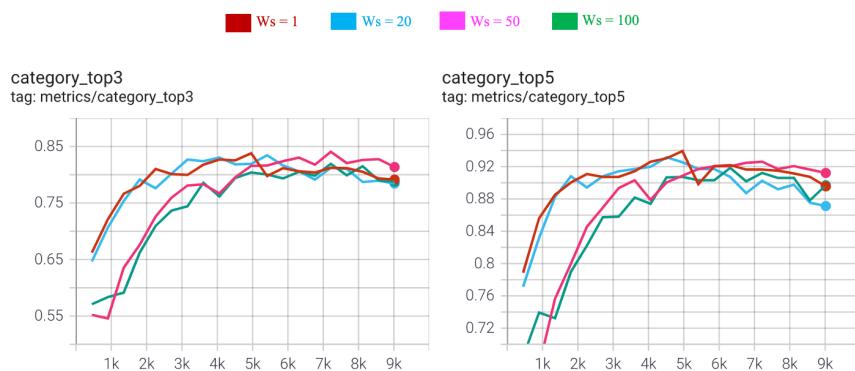




图 7.4 不同 Ws 取值对应的验证集类别预测 Top3 和 Top5 准确率

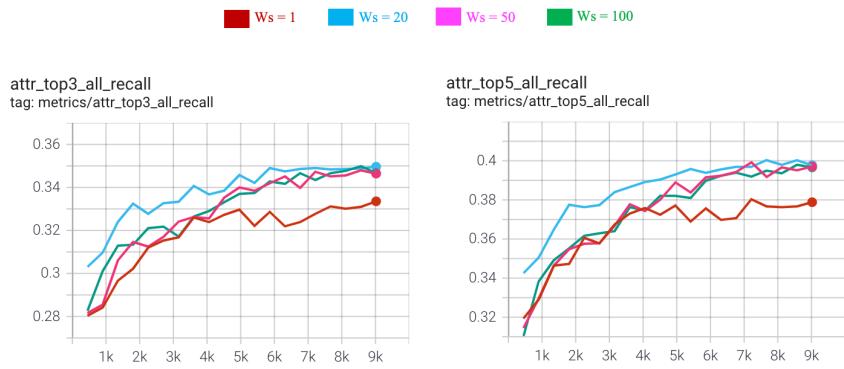


图 7.5 不同 Ws 取值对应的验证集属性预测 Top3 召回率和 Top5 召回率

所以，在之后的训练设置中，本文均取 $W_c = 1$ ， $W_s = 20$ 。后续将不会改变该超参数的取值。

7.7 Cloth Resnet A 相关实验

7.7.1 迭代次数 epoch 实验

在神经网络训练时，epoch 代表神经网络迭代更新参数的轮次。如果 epoch 太小，则网络无法有足够的步长达到收敛点。如果 epoch 太大，则网络会面临“过拟合”问题，即网络过于符合训练集中的数据规律，反而无法在验证集上取得很好的效果。在本文中，确认 epoch 采用如下思路：设定一个很小的学习率训练 100 个 epoch，观察训练集分类准确率和属性预测召回率的预测情况，测得到达收敛时的 epoch 值，记作 lim_epoch 。因为低学习率对应的参数变化步长很低，所以如果模型能在低学习率时以 lim_epoch 达到收敛，那么模型在使用高学习率训练时，必然可以在更少的 epoch 内更快到达收敛（但是也有可能在收敛点附近震荡）。所以，在后续关于学习率的实验时，均采用该 lim_epoch 值，这样可以保证实验中不会出现训练完成仍未到达收敛点的情况。所以设置如下实验：采用学习率 $learning\ rate = 0.00005(5e-5)$ ，衰减速率 $learning\ rate\ decay = 1$ ，分类权重 $W_c = 1$ ，属性权重 $W_s = 20$ ，训练 100 个 epoch，观察 Cloth Resnet A 在训练集上的收敛情况。

7.7.1.1 实验结果与分析

Cloth Resnet A 训练集分类 Topn 准确率和训练集属性预测 Topn 召回率随迭代次数的变化曲

线如图所示：

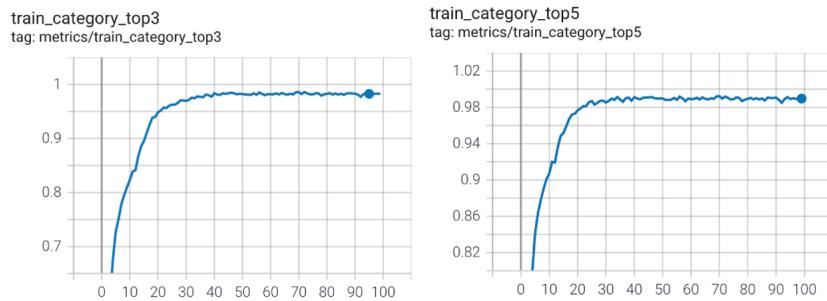


图 7.6 Cloth Resnet A 在 100 个 epoch 下的训练集分类准确率变化情况

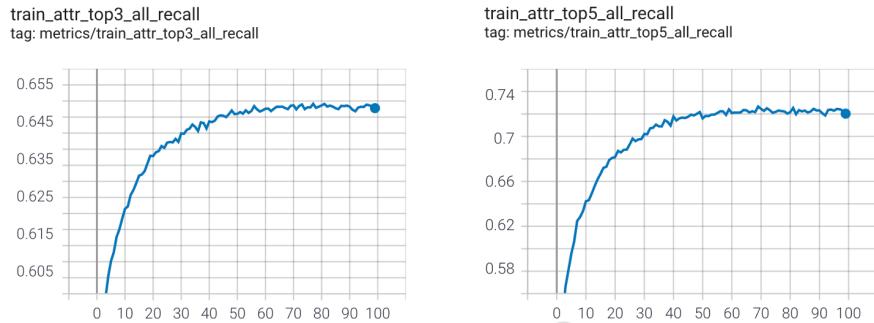


图 7.7 Cloth Resnet A 在 100 个 epoch 下的训练集属性预测召回率变化情况

根据以上实验结果可知，在学习率很小时，分类准确率在约 40 个 epoch 即达到收敛。而属性预测召回率在约 65 个 epoch 达到收敛。因此，可以确定 lim_epoch 的值取 65 为宜。

7.7.2 学习率 learning rate 和衰减速率 learning rate decay 实验

根据 Adam 优化算法论文[30]建议，Adam 优化器使用的学习率应在 0.001~0.0001 之间。同时，更小的学习率应该更小同的衰减速率，保证模型以较小的步长调整对应参数。因此，为了获得最好的性能，本实验设置三个不同的学习率和对应衰减速率，分别为 0.001/0.9, 0.0005/0.95, 0.0001/1 其余参数均相等，即 $\text{epoch} = 60$, $\text{Wc} = 1$, $\text{Ws} = 20$ ，统计训练后的验证集网络性能，并记录最优秀的网络性能。

7.7.2.1 实验结果与分析

三种不同学习率下 Cloth Resnet A 在验证集上的网络性能统计如下：

装
订
线



图 7.8 Cloth Resnet A 学习率实验结果

可以看到，在学习率为 0.0005/0.9 时，Cloth Resnet A 可以取得最优秀的性能。在此，可以统计得到 Cloth Resnet A 最佳性能准确率、召回率以及训练参数如下表所示：

参数/评价指标名	值
迭代次数(epoch)	26
学习率(learning rate)	5e-4 (0.0005)
学习率衰减常数(learning rate decay)	0.95
分类权重(Wc)	1
属性预测权重(Ws)	20
分类 Top3 准确率	83.8%
分类 Top5 准确率	92.1%
属性预测 Top3 总召回率	41.6%
属性预测 Top3 召回率——纹理(Texture)	46.0%
属性预测 Top3 召回率——形状(Shape)	47.9%
属性预测 Top3 召回率——布料(Fabric)	35.3%
属性预测 Top3 召回率——组成部分(part)	35.7%
属性预测 Top3 召回率——风格(style)	48.2%
属性预测 Top5 总召回率	49.1%
属性预测 Top5 召回率——纹理(Texture)	53.5%
属性预测 Top5 召回率——形状(Shape)	53.7%
属性预测 Top5 召回率——布料(Fabric)	45.2%
属性预测 Top5 召回率——组成部分(part)	41.1%
属性预测 Top3 召回率——风格(style)	52.6%

表 7.1 Cloth Resnet A 最佳参数设置与对应性能

装
订
线

7.8 Cloth Resnet B 相关实验

由于 Cloth Resnet A 与 Cloth Resnet B 的网络结构基本相同，仅最后的分类链接层存在差异，所以对 Cloth Resnet B 采用与 Cloth Resnet A 相同的 $\text{lim_epoch} = 65$ ，因此不再进行迭代次数实验。仅进行与 Cloth Resnet A 相同的学习率和衰减速率实验。

7.8.1 学习率 learning rate 和衰减速率 learning rate decay 实验

为了获得 Cloth Resnet B 最好的性能，实验设置三个不同的学习率和对应衰减速率，分别为 0.001/0.9, 0.0005/0.95, 0.0001/1 其余参数均相等，即 $\text{epoch} = 60$, $\text{Wc} = 1$, $\text{Ws} = 20$ ，统计训练后的验证集网络性能，并记录最优秀的网络性能。

7.8.1.1 实验结果与分析

装
订
线

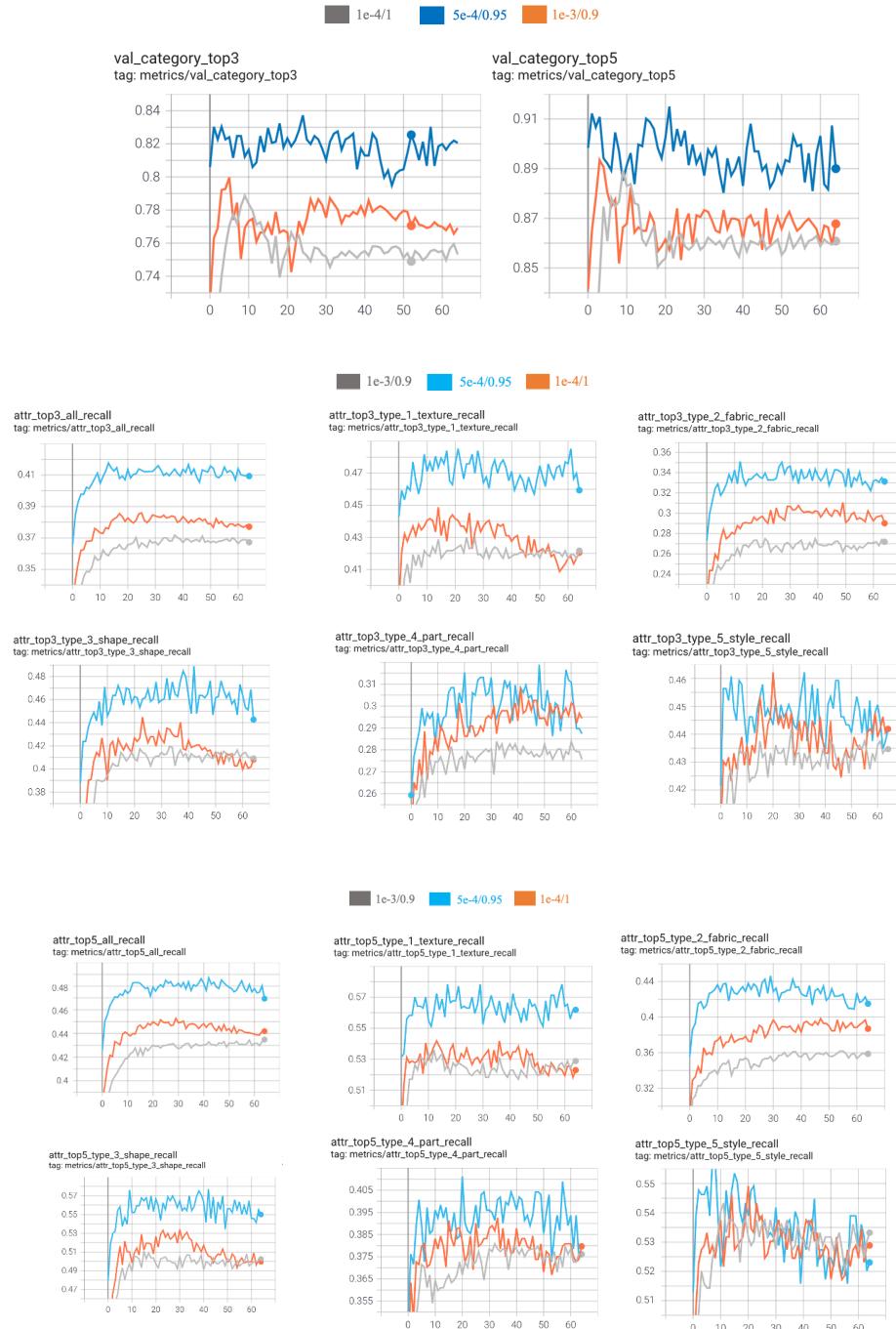


图 7.9 Cloth Resnet B 学习率实验结果

可以看到，在学习率为 0.0005/0.9 时，Cloth Resnet B 可以取得最优秀的性能。在此，可以统计得到 Cloth Resnet B 最佳性能准确率、召回率以及训练参数如下表所示：

参数/评价指标名	值
迭代次数(epoch)	28

学习率(learning rate)	5e-4 (0.0005)
学习率衰减常数(learning rate decay)	0.95
分类权重(Wc)	1
属性预测权重(Ws)	20
分类 Top3 准确率	83.5%
分类 Top5 准确率	91.2%
属性预测 Top3 总召回率	41.3%
属性预测 Top3 召回率——纹理(Texture)	48.2%
属性预测 Top3 召回率——形状(Shape)	47.2%
属性预测 Top3 召回率——布料(Fabric)	35.1%
属性预测 Top3 召回率——组成部分(part)	31.2%
属性预测 Top3 召回率——风格(style)	45.1%
属性预测 Top5 总召回率	48.2%
属性预测 Top5 召回率——纹理(Texture)	57.3%
属性预测 Top5 召回率——形状(Shape)	56.8%
属性预测 Top5 召回率——布料(Fabric)	43.8%
属性预测 Top5 召回率——组成部分(part)	40.5%
属性预测 Top3 召回率——风格(style)	52.1%

表 7.2 Cloth Resnet B 最佳参数设置与对应性能

7.9 Cloth Rep VGG 训练设置

7.9.1 训练迭代次数 epoch 实验

为了获得 Cloth Rep VGG 训练的 lim_epoch, 采用类似 7.4.2 的实验设置。采用学习率 learning rate = 0.00005(5e-5), 衰减速率 learning rate decay = 1, 分类权重 Wc = 1, 属性权重 Ws = 20, 训练 100 个 epoch, 观察网络在训练集上的拟合情况。

7.9.1.1 实验结果与分析

Cloth Rep VGG 在训练集上的拟合情况如下所示:

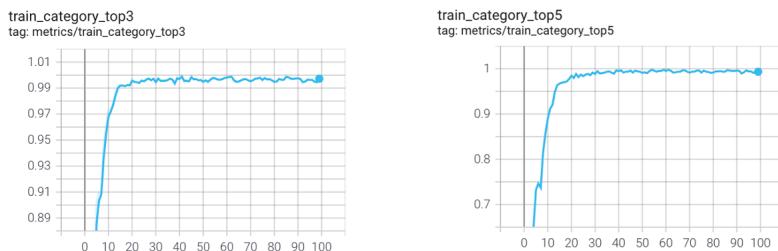


图 7.10 Cloth Rep VGG 在 100 个 epoch 下的训练集分类准确率变化情况

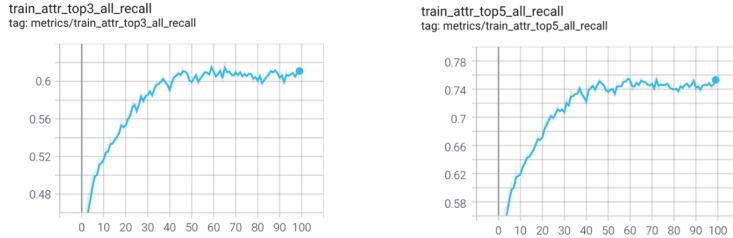


图 7.11 Cloth Rep VGG 在 100 个 epoch 下的训练集属性预测召回率变化情况

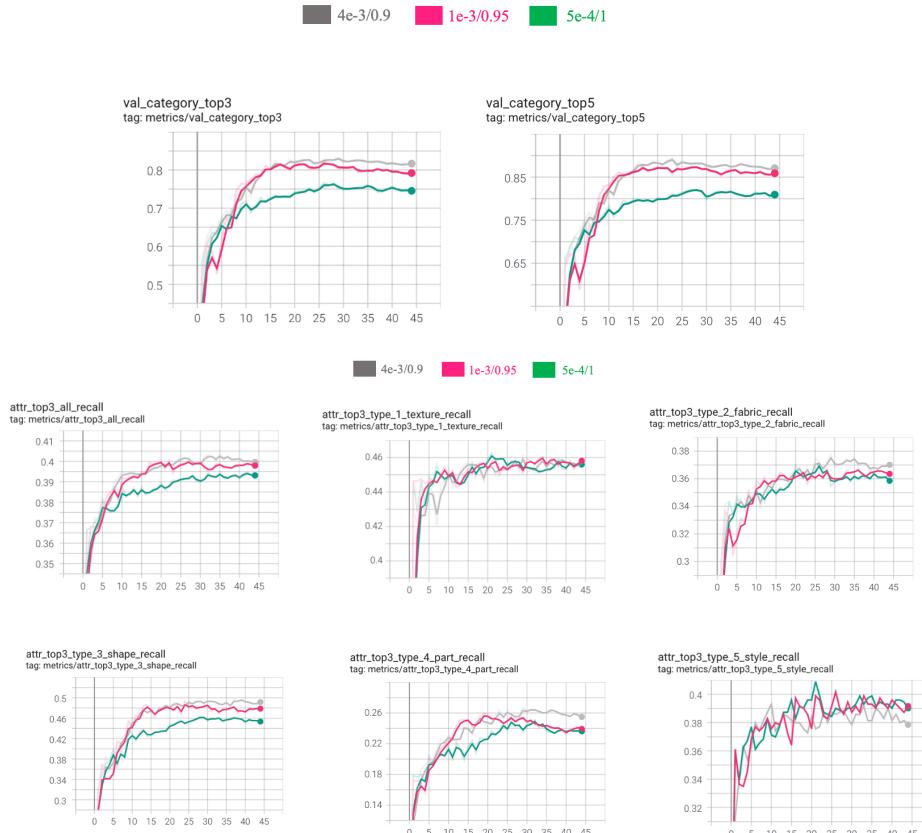
根据以上实验结果可知，在学习率很小时，分类准确率在约 25 个 epoch 即达到收敛。而属性预测召回率在约 45 个 epoch 达到收敛。因此，可以确定 lim_epoch 的值取 45 为宜。

7.9.2 学习率 learning rate 和衰减速率 learning rate decay 实验

在上一个实验中，已经获得了 Cloth Rep VGG 训练应使用的 epoch。为了获得 Cloth Rep VGG 训练的最佳学习率设置，进行类似 7.4.3 学习率和衰减速率实验。由于论文[26]中的示例采用的是 0.004 的学习率，所以实验设置于 resnet 实验不同的三个学习率和对应衰减速率，分别为 0.004/0.9, 0.001/0.95, 0.0005/1 其余参数均相等，即 epoch = 30, Wc = 1, Ws = 20, WI = 2000，统计网络在验证集下的性能，并记录最佳性能。

7.9.2.1 实验结果与分析

Cloth Rep VGG 在验证集的拟合情况如下所示：



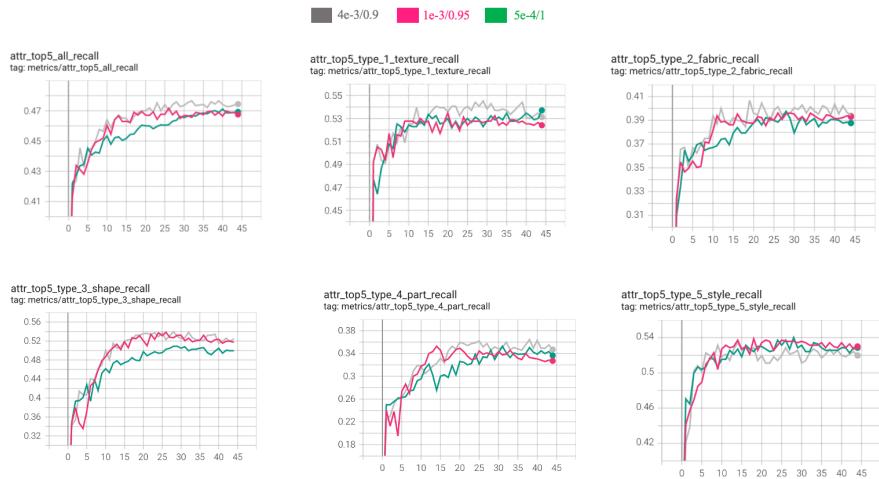


图 7.12 Cloth Rep VGG 学习率实验结果

可以看到，在学习率为 0.004/0.9 时，Cloth Rep VGG 可以取得最优秀的性能。在此，可以统计得到 Cloth Resnet B 最佳性能准确率、召回率以及训练参数如下表所示：

装
订
线

参数/评价指标名	值
迭代次数(epoch)	33
学习率(learning rate)	4e-3(0.004)
学习率衰减常数(learning rate decay)	0.9
分类权重(Wc)	1
属性预测权重(Ws)	20
分类 Top3 准确率	82.7%
分类 Top5 准确率	88.9%
属性预测 Top3 总召回率	40.3%
属性预测 Top3 召回率——纹理(Texture)	45.6%
属性预测 Top3 召回率——形状(Shape)	47.4%
属性预测 Top3 召回率——布料(Fabric)	37.2%
属性预测 Top3 召回率——组成部分(part)	27.1%
属性预测 Top3 召回率——风格(style)	39.2%
属性预测 Top5 总召回率	47.3%
属性预测 Top5 召回率——纹理(Texture)	54.5%
属性预测 Top5 召回率——形状(Shape)	54.1%
属性预测 Top5 召回率——布料(Fabric)	40.5%
属性预测 Top5 召回率——组成部分(part)	36.3%
属性预测 Top3 召回率——风格(style)	52.3%

表 7.3 Cloth Rep VGG 最佳参数设置与对应性能

7.10 Landmark 预测权重 Wl 实验

由于 Fashion Net 在网络结构上引入了预测网络边界点的子网络，所以 Fashion Net 需要在之前的损失函数中加入边界值损失项 $L_{landmark}$ (详见 5.3)。因此，Fashion Net 会引入新的参数 W_l ，它代表边界值点损失项的权重。经过代码测算， W_l 的初始值在 $L_{landmark}$ 在(0.005~0.01)区间。依照 7.4.1 中的思路， $L_{landmark}$ 的数量级应当尽量与其他其他数量级保持一致。为了确定 W_l ，本文设置如下实验：将 W_c 的值固定为 1， W_s 的值固定为 20，分别取 W_l 的值为[400, 800, 2000]，均以 0.001 的 learning rate，0.9 的 learning decay rate，训练 20 个 epoch，观察模型损失变化情况和验证集性能。

7.10.1 实验结果与分析

该实验的实验结果如下，如下是设置不同的 W_l 后四种 loss 的取值情况：

装
订
线

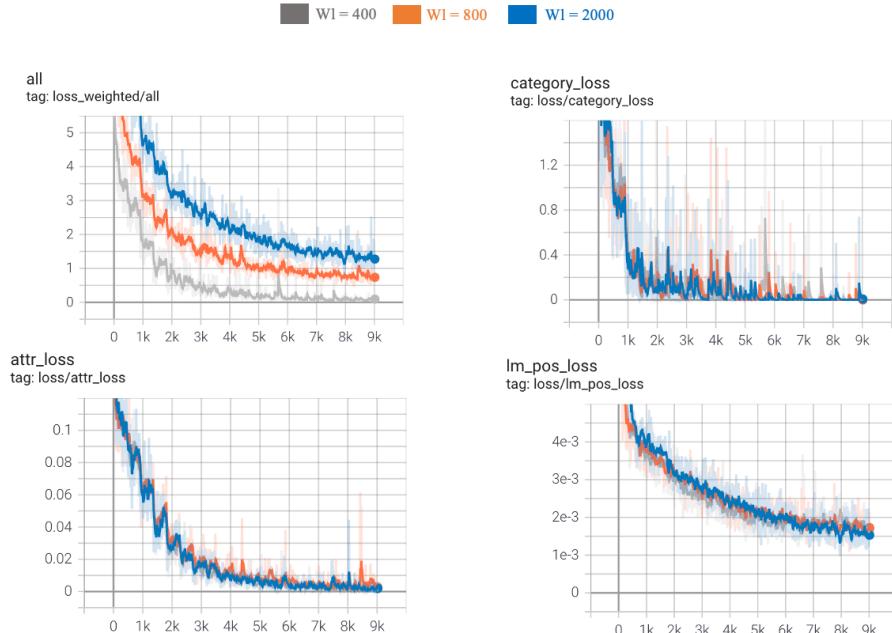


图 7.13 Fashion Net 改变 W_l 后模型损失下降情况

可以看到，尽管 W_l 的数量级发生了改变，但是每种损失下降曲线非常相似。可见 W_l 数量级对模型的训练过程的影响与 W_s 和 W_c 不同，它基本不会影响模型的损失下降速率。

如下是设置不同 W_l 后的训练集评估数值变化和验证集评估数值变化：

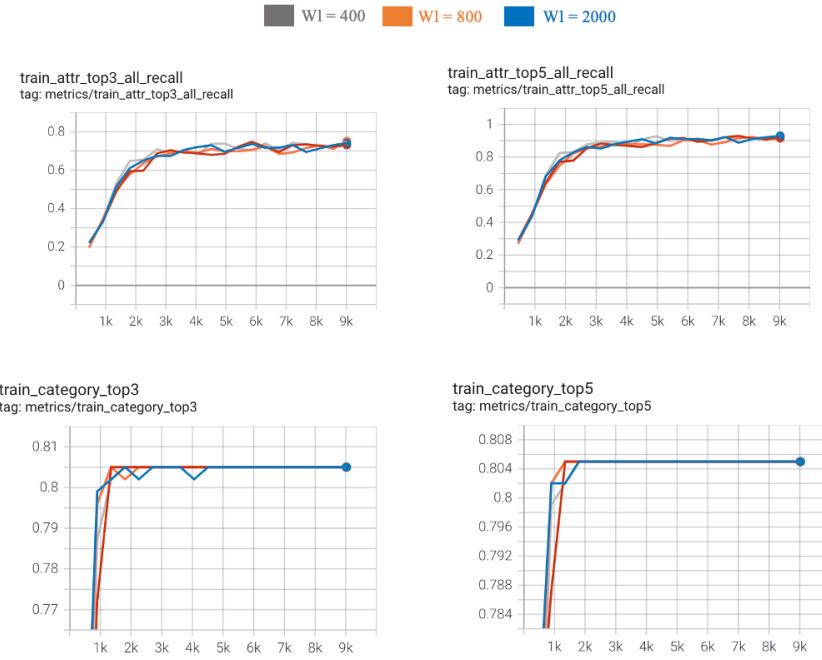
装
订
线

图 7.14 改变 W1 后的训练集分类准确率和属性预测召回率

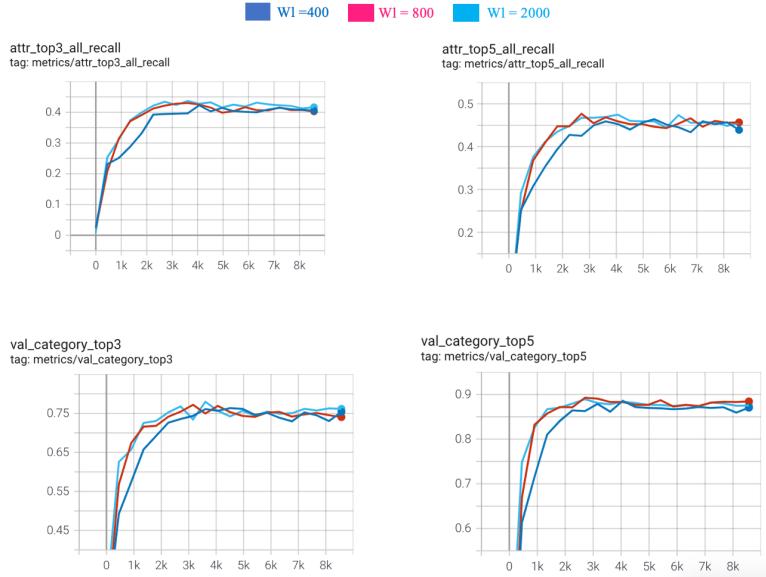


图 7.15 Fashion Net 改变 W1 后的验证集分类准确率和属性预测召回率

通过实验结果可以发现，在改变 W1 后，训练集的拟合曲线基本没有变化，但是在验证集上， $w1 = 2000$ （对应浅蓝色曲线）能够取稍优秀的性能。分析网络结构可以发现，W1 对应的是 landmark prediction branch 子网络的网络参数更新，该子网络与分类和属性预测网络的共享参数非常少，并且会经过差异很大的卷积过程。论文[33]中提到，这是一种软参数链接(Soft-parameter-sharing) 的结构。对于软参数链接，不同的 loss 子项之间的影响非常小。并且，采用软参数链接时，由于总损失对另一子网络的偏导近似为 0，训练过程非常类似于训练两个独立的网

络。这就导致 $W1$ 的取值不会严重影响分类和属性网络的性能。对于一个独立的网络而言，Adam 优化器的优化方案会自动消除 loss 数量级带来的影响。根据 Adam 优化器的参数更新公式：

$$\theta_t = \theta_{t-1} - \alpha \cdot \frac{\widehat{m}_t}{\sqrt{\widehat{v}_t} + \epsilon}$$

其中 \widehat{m}_t 为动量的一阶矩估计，它由梯度 g_t 累加而成。 \widehat{v}_t 为动量的二阶矩估计，它由梯度的平方 g_t^2 累加而成。现在假设将损失扩大 δ 倍，那么其对应的梯度也会增加 δ 倍。将其带入公式有：

$$\theta_t' = \theta_{t-1} - \alpha \cdot \frac{\mu \widehat{m}_t}{\sqrt{\mu^2 \widehat{v}_t} + \epsilon}$$

由于 ϵ 只是一个很小的偏置常数，在实际运算中不会对分子分母产生大量影响，所以 μ 可以近似看作可以约分。即有 $\theta_t' \approx \theta_{t-1}$ 。由此可见，在单任务模型中，Adam 优化器可以极大降低 loss 损失的数量级对参数变化的影响。所以， $W1$ 的取值也不会显著影响 landmark prediction

但是，可以看到在验证集上， $W1$ 取值为 2000 的模型性能会有稍微更优秀的性能。在此对此现象作出两种推测。第一是 $W1 = 2000$ 时对应的边界值预测子网络性能更加优秀。可以看到在图表 29 中的边界点损失的最后收敛水平中，由于 $W1 = 2000$ 收敛到了最低的位置，说明 $W1 = 2000$ 时，边界点预测网络的性能更优秀，因此体现在测试集准确率上 $W1 = 2000$ 准确率更高。第二是实验采用的测试集规模不足，由于 Fashion Net 的网络结构非常复杂，其原本实验环境为 289,222 张图片，而实验时将其缩减到了 200,000 张图片，数据集的缩减这导致训练集规模无法很好地表述验证集规律，最终导致验证集准确率出现了波动。其中第二种推测的可能性更大，因为实验已经证明 $W1$ 的取值在训练集的拟合上并不会带来明显影响。

但是 $W1$ 的取值必须保证与其他的损失在相近的数量级。如下是取 $W1 = 20000$ 时的训练集属性预测召回率和分类准确率：

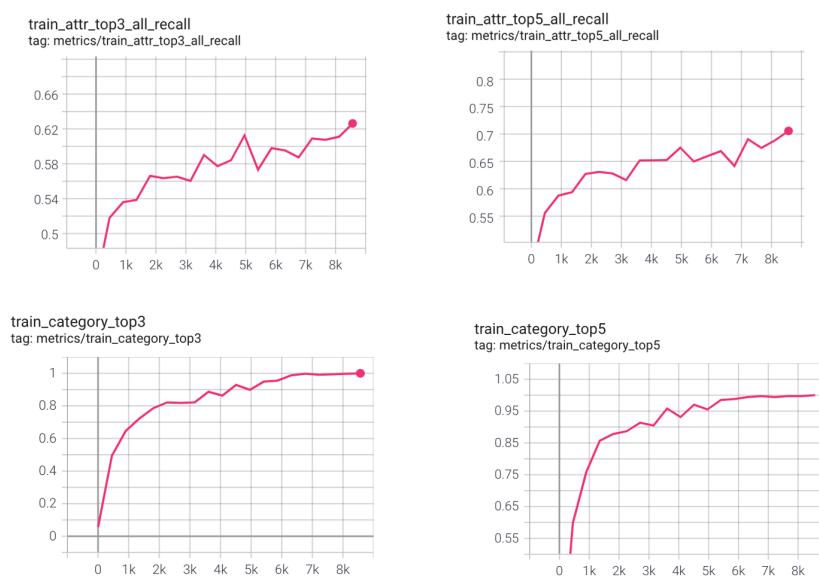


图 7.16 $W1 = 20000$ 时的训练集属性预测召回率与分类准确率

可以看到，在 $W1 = 20000$ 时，由于边界值损失的数量级已经远大于分类和属性任务的数量级，分类任务和属性预测任务的收敛速度都明显变慢。

所以只要保持分类损失、边界值损失和属性预测损失在相似的数量级时， $W1$ 的取值并不会严重影响模型性能，在后续的实验中，采用 $W1 = 2000$ 。原因如下：(1) 本实验证明 $W1 = 2000$ 时模型的训练集拟合程度非常好，而验证集的准确率也有优势。(2) 取 $W1 = 2000$ ，可以增大模型损失数量级，便于之后的实验数据统计。

7.11 Fashion Net 相关实验

7.11.1 训练迭代次数 epoch 实验

为了获得 Fashion Net 训练的 lim_epoch ，采用类似 7.4.2 的实验设置。采用学习率 $learning rate = 0.00005(5e-5)$ ，衰减速率 $learning rate decay = 1$ ，分类权重 $Wc = 1$ ，属性权重 $Ws = 20$ ，训练 100 个 epoch，观察网络在训练集上的拟合情况。

7.11.1.1 实验结果与分析

装
订
线

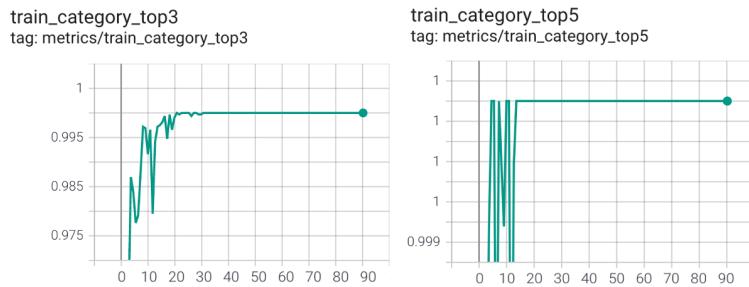
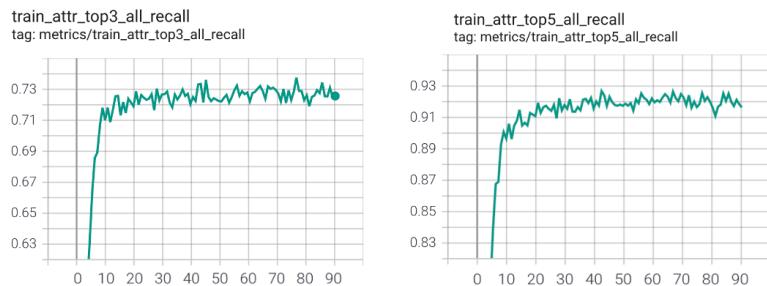


图 7.17 Fashion Net 在 100 个 epoch 下的训练集属性预测召回率变化情况



图表 7.18 Fashion Net 在 100 个 epoch 下的训练集属性预测召回率变化情况

根据以上实验结果可知，由于 Fashion Net 提供了预训练参数，所以属性预测准确率和分类准确率均在 30 个 epoch 便达到收敛。因此，可以确定 lim_epoch 的值取 30 为宜。

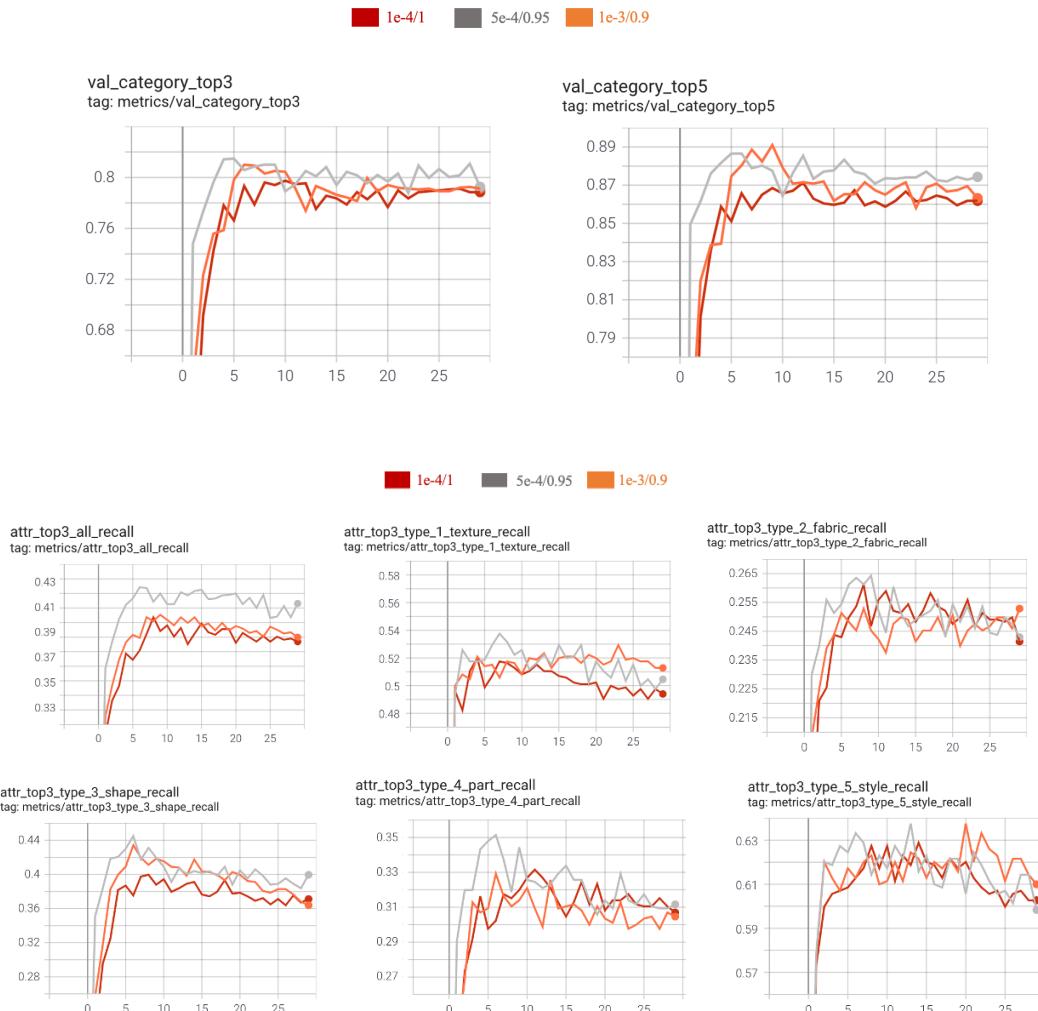
7.11.2 学习率 learning rate 和衰减速率 learning rate decay 实验

在上一个实验中，已经获得了 Fashion Net 训练应使用的 epoch。为了获得 Fashion Net 的训练的最佳学习率设置，进行类似 7.4.3 学习率和衰减速率实验。实验设置三个不同的学习率和对应衰减速率，分别为 $0.001/0.9, 0.0005/0.95, 0.0001/1$ 其余参数均相等，即 $epoch = 30, Wc = 1, Ws$

= 20, W1 = 2000, 统计网络在验证集下的性能，并记录最佳性能。

7.11.2.1 实验结果与分析

Fashion Net 在不同学习率与衰减速率下的验证集性能如下所示：



装
订
线

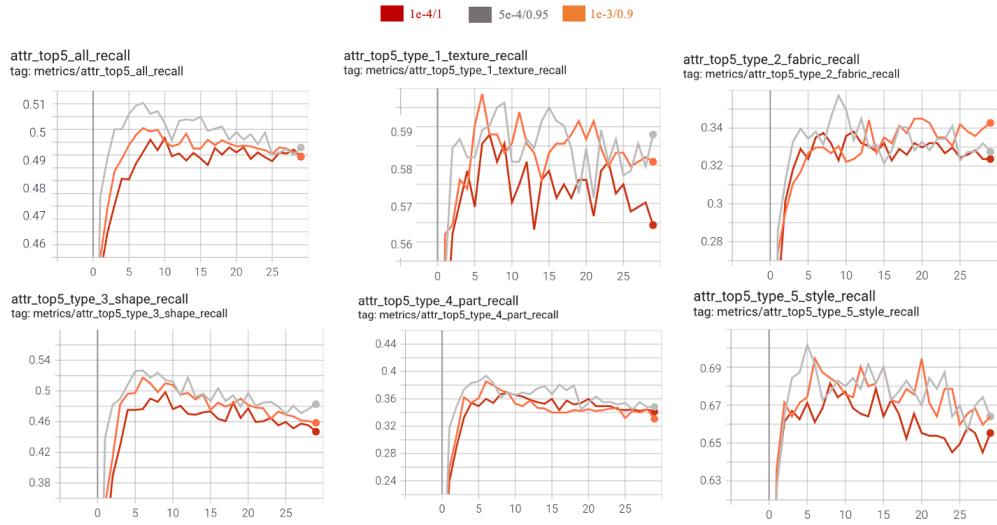


图 7.19 Fashion Net 学习率实验结果

可以看到，在学习率为 0.0005/0.9 时，Fashion Net 可以取得最优秀的性能。在此，可以统计得到 Fashion Net 最佳性能准确率、召回率以及训练参数如下表所示：

参数/评价指标名	值
迭代次数(epoch)	8
学习率(learning rate)	5e-4 (0.0005)
学习率衰减常数(learning rate decay)	0.95
分类权重(Wc)	1
属性预测权重(Ws)	20
分类 Top3 准确率	80.6%
分类 Top5 准确率	88.9%
属性预测 Top3 总召回率	42.7%
属性预测 Top3 召回率——纹理(Texture)	53.9%
属性预测 Top3 召回率——形状(Shape)	45.8%
属性预测 Top3 召回率——布料(Fabric)	27.3%
属性预测 Top3 召回率——组成部分(part)	36.1%
属性预测 Top3 召回率——风格(style)	63.2%
属性预测 Top5 总召回率	50.9%
属性预测 Top5 召回率——纹理(Texture)	60.1%
属性预测 Top5 召回率——形状(Shape)	52.8%
属性预测 Top5 召回率——布料(Fabric)	35.6%
属性预测 Top5 召回率——组成部分(part)	38.7%
属性预测 Top3 召回率——风格(style)	70.2%

表 7.4 Fashion Net 最佳参数设置与对应性能

7.12 New Fashion Net 相关实验

New Fashion Net 一共整合了三种优化结构，分别包括：在全局特征提取网络中采用 Resnet 框架替代 VGG-16 框架，在边界点预测采用 Rep-VGG 框架替代 VGG-16 框架，在损失函数中采用 SSIM 损失替代交叉熵损失。在实验时，本文首先会对 New Fashion Net 的整体性能展开实验。之后会设计实验依次验证每一部分的性能提升。

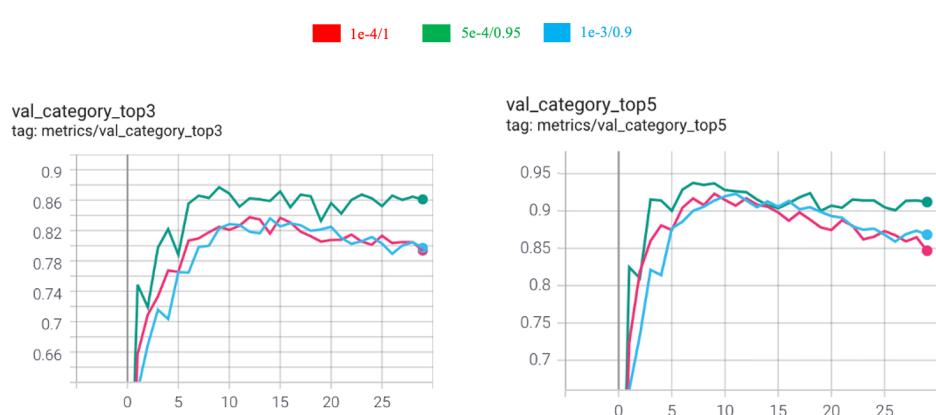
New Fashion Net 是通过替换 Fashion Net 的骨架网络得到，而之前的实验已经证明，在实际训练时，验证集上 Fashion Net 在 epoch = 8 附近达到收敛，Resnet50 在 epoch = 25 附近到达收敛，Rep VGG 在 epoch = 33 附近达到收敛。但是在 landmark prediction branch 中仅引入了 6 个 Repblock。综合以上条件推断，对 New Fashion Net 开展实验时，取 epoch = 30 便可以确保网络达到收敛。考虑到 New Fashion Net 庞大的训练时间开销，在此不再进行迭代次数实验，直接根据推论使用 epoch = 30。

7.12.1 学习率 learning rate 和衰减速率 learning rate decay 实验

为了获得 New Fashion Net 的训练的最佳学习率设置，进行类似 7.4.3 学习率和衰减速率实验。实验设置三个不同的学习率和对应衰减速率，分别为 0.001/0.9, 0.0005/0.95, 0.0001/1 其余参数均相等，即 epoch = 30, Wc = 1, Ws = 20, WI = 2000，统计网络在验证集下的性能，并记录最佳性能。

7.12.1.1 实验结果与分析

New Fashion Net 在不同学习率与衰减速率下的验证集性能如下所示：



装
订
线

图 3.20 New Fashion Net 学习率实验结果

在 New Fashion Net 中，存在分类和属性预测问题收敛点不一样的问题。其中，分类点在 $\text{epoch} = 8$ 左右即收敛，而属性预测在 $\text{epoch} = 25$ 左右才达到收敛。权衡性能，本文选择 $\text{epoch} = 18$ 为 New Fashion Net 的收敛点。此时，分类和属性预测均能取得较为优秀的性能。可以看到，在学习率为 0.0005/0.9 时，New Fashion Net 可以取得最优秀的性能。在此，可以统计得到 Fashion Net 最佳性能准确率、召回率以及训练参数如下表所示：

参数/评价指标名	值
迭代次数(epoch)	8
学习率(learning rate)	5e-4 (0.0005)
学习率衰减常数(learning rate decay)	0.95

分类权重(Wc)	1
属性预测权重(Ws)	20
边界点损失(Wl)	2000
分类 Top3 准确率	87.9%
分类 Top5 准确率	92.8%
属性预测 Top3 总召回率	49.6%
属性预测 Top3 召回率——纹理(Texture)	52.7%
属性预测 Top3 召回率——形状(Shape)	46.9%
属性预测 Top3 召回率——布料(Fabric)	43.1%
属性预测 Top3 召回率——组成部分(part)	39.2%
属性预测 Top3 召回率——风格(style)	66.6%
属性预测 Top5 总召回率	53.2%
属性预测 Top5 召回率——纹理(Texture)	59.1%
属性预测 Top5 召回率——形状(Shape)	55.6%
属性预测 Top5 召回率——布料(Fabric)	52.5%
属性预测 Top5 召回率——组成部分(part)	43.8%
属性预测 Top3 召回率——风格(style)	72.1%

表 7.5 New Fashion Net 最佳参数设置及对应性能

7.12.2 New Fashion Net 各部分优化效果实验

由于New Fashion Net 一共包含了三种网络结构修改优化技术，在本实验中，将分别实现其中一个优化部分然后比较各个优化部分的性能。三个网络的训练参数均为 epoch = 30, Wc = 1, Ws = 20, Wl = 2000。统计不同网络在验证集下的性能。

同时，实验还对 SSIM 损失函数对模型的边界值损失 $L_{landmark}$ 和边界值预测结果误差进行了统计。这里的边界值预测结果误差使用预测点位置到实际点位置的欧式距离平均值来表示，即：

$$Dist = \sum_{i=1}^8 ||\mathbf{x}' - \mathbf{x}||^2 \quad (7.4)$$

其中， $Dist$ 代表边界点误差，其中 \mathbf{x}' 代表预测点位置， \mathbf{x} 代表实际点位置。

7.12.2.1 实验结果与分析

New Fashion Net 三种优化技术为 Fashion Net 带来的优化效果统计如下：

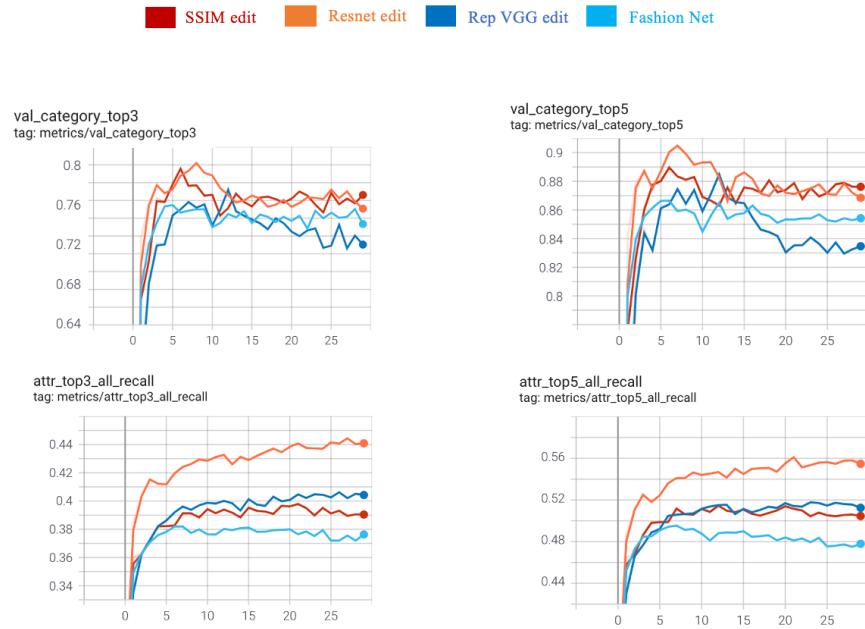


图 7.21 New Fashion Net 各部分优化效果对比

根据实验结果可以分析得到，将 Fashion Net 的框架更换为 Resnet 可以同时的提高模型的属性标签预测能力和分类预测能力。将损失函更换为 SSIM 也可以同时提高模型的预测能力和分类准确率。但是将模型的 landmark prediction branch 自网络替换为 Rep VGG 后，可以提高模型的标签预测能力，但是不能提高模型的分类能力。

对 SSIM 损失函数对模型的边界值损失 $L_{landmark}$ 和边界值预测结果误差的统计结果如下：

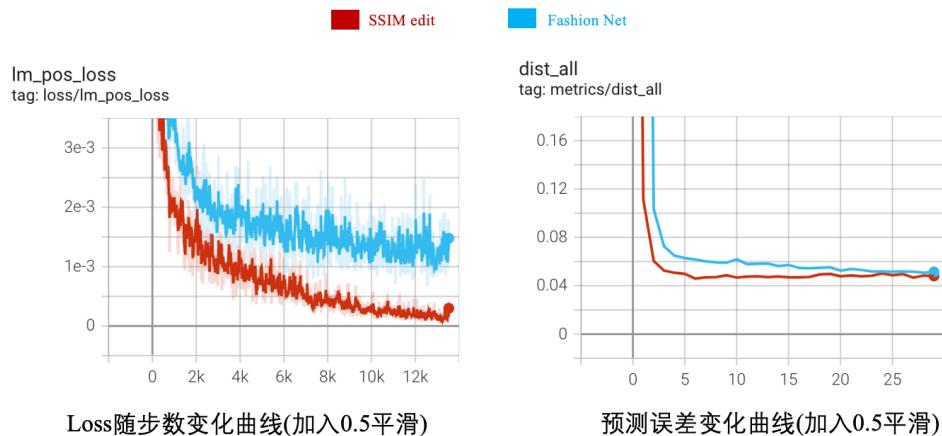


图 7.22 加入 SSIM 后的边界点预测性能对比

根据实验结果可以看出，SSIM 的加入可以让边界点损失更快的收敛并改变 $L_{landmark}$ 的数量级。但是 SSIM 损失只能提高训练速度，最后还是会与 MSELoss 收敛到近乎相同的精度。

7.13 模型时效性能对比

本文还将不同模型的参数量和平均迭代时间进行了统计。平均迭代时间为在 GTX2080 设备上完成一个 epoch 需要的训练时间。其统计结果如下：

模型名称	参数量	平均迭代时间(min)
CLOTH RESNET A	29,753,408	5.16
CLOTH RESNET B	40,506,432	5.83
CLOTH REP VGG	11,189,712	1.38
FASHION NET	76,298,024	6.56
NEW FASHION NET	108,154,640	7.63

表 7.6 模型的参数量和平均迭代时间表

采用 matplotlib 对参数量和平均迭代时间进行可视化如下所示：

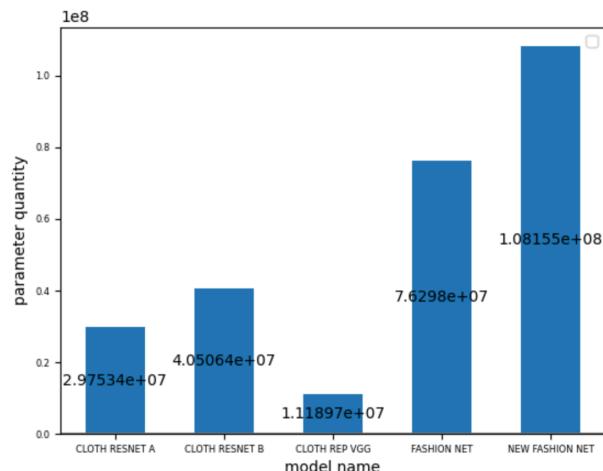


图 7.23 模型参数量统计图

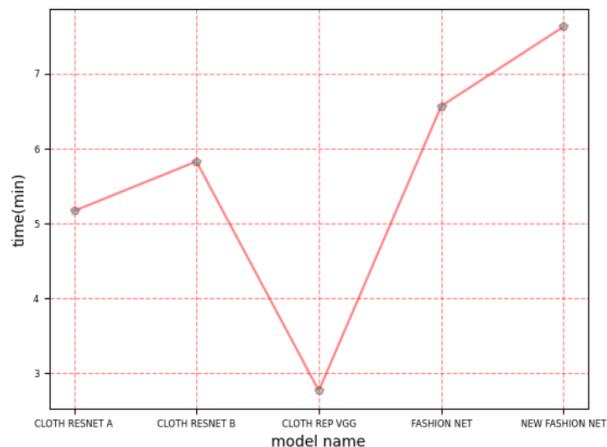


图 7.24 模型平均迭代时间统计图

其中，参数量反映了模型在训练时的显存占用率，即训练空间开销；而平均迭代时间反应了模型的训练时间开销。两者共同反应了网络结构的复杂程度。

装
订
线

8 结论和展望

8.1 结论

本文创新地提出用深度学习中的多任务卷积神经网络来完成服装图像自动标注任务，设计了共4种网络结构，并开展了综合性试验来对比它们的性能。下面是文章的几种网络结构与几个著名的服装图片自动标注参考文献的具体性能对比(Top3/Top5)：

	分类准确率 (%)	全属性召回 率(%)	纹理召回率 (%)	形状召回率 (%)	布料召回率 (%)	组成部召回 率(%)	风格召回率 (%)
CLOTH RESNET A	83.8/92.1	41.6/49.1	46.0/53.5	47.9/53.7	35.3/45.2	36.9/41.1	48.2/52.6
CLOTH RESNET B	83.5/91.2	41.3/48.2	48.2/57.3	47.2/56.8	35.1/43.8	36.2/40.5	45.1/52.1
CLOTH REP VGG	82.7/88.9	40.3/47.3	45.6/54.5	47.4/54.1	37.2/40.5	27.1/36.3	39.2/52.3
FASHION NET (exp)	80.6/88.9	42.7/50.9	53.9/60.1	45.8/52.8	27.3/35.6	36.1/38.7	63.2/70.2
NEW FASHION NET	87.9/92.8	49.6/53.2	52.7/59.1	46.9/55.6	43.1/52.5	39.2/43.8	66.6/72.1
WTBI[19]	43.73/66.26	27.46/35.37	24.21/32.65	23.39/31.26	25.38/36.06	26.31/33.24	49.85/56.68
DARN[21]	59.48/79.58	36.15/48.15	36.15/48.15	35.89/46.93	36.64/48.52	39.17/50.14	66.11/73.16
Lu.et.al[36]	86.72/92.51	--	--	--	--	--	--
Wang.et.al[37]	90.99/95.78	51.53/60.95	50.31/65.48	53.32/61.05	40.31/48.23	40.65/56.32	68.70/74.25
Fashion Net[14]	82.58/90.17	45.52/54.61	56.17/61.83	50.10/59.50	39.30/49.84	44.13/54.02	66.43/73.16

根据实验结果，可以得到如下关于网络架构的结论：

- (1) 网络拟合能力 Resnet > Rep VGG50 > VGG -16。通过实验结果的分类准确率可以看到，因为尽管实验中 Cloth Resnet A、Cloth Resnet B、Cloth 没有加入可以提取局部特征的子网络，但是其性能仍然大于实验中的 Fashion Net。
- (2) 复杂化 Resnet 的分类链接层并不能优化网络性能。虽然 Cloth Resnet B 在属性分类的链接层中加入了类似 VGG 的 3*3 卷积块复杂了网络结构，但是其性能并不能超过 Cloth Resnet A。分析原因其一可能是因为本来 Resnet50 前面的特征提取层采用残差学习的结构，新加入没有残差块的纯卷积结构会恶化残差网络性能。其二，更复杂的网络结构容易引起过拟合问题，导致网络的性能变差。
- (3) 局部特征子网络对属性预测的优化非常明显。可以看到 Fashion Net 在在属性预测方面的性能明显优于没有引入局部特征子网络的网络。

根据实验结果和 7.9 中的模型性能统计，本文对 5 个网络的性能进行如下综合评述：

装
订
线

- (1) Cloth Resnet A: 该模型分类能力出色，属性预测能力由于没有引入局部特征导致一般，网络训练显存开销和时效开销中等。
- (2) Cloth Resnet B: 该模型的整体性能与 Cloth Resnet A 相近，但是训练显存开销和时效开销更大，不推荐使用。
- (3) Cloth Rep VGG: 该模型的整体性能稍逊色于 Cloth Resnet A，但是训练显存开销和时销开销很低，适合在需要快速训练模型，但是对精度要求不严格的情况使用。
- (4) Fashion Net: 该模型的属性预测能力出色，但是分类能力中等。由于引入识别边界值的子网络，训练显存开销和时效开销高。
- (5) New Fashion Net: 该模型的属性预测能力和分类能力都很优秀。在压缩到 200,000 张的数据集上训练的分类和属性预测表现都超过了在 200,000 张数据集上进行训练的 Fashion Net，并且达到逼近基线数据 Wang.et.al 设计的网络的性能。New Fashion Net 对布料召回率的提升尤为明显，在 200,000 张数据集上已经可以超越基线数据的 Top3 属性预测准确率 3%，Top5% 属性预测准确率 5%。但是该模型的训练显存开销和时效开销非常高。该模型适合在有大量训练资源，追求高精度服装图像自动标注时使用。

8.2 展望

针对本文在实验中遇到的问题和相关工作总结，以下对未来的研究方向做一个简单的展望。

8.2.1 数据增广

在本文实验中，由于实验设备的设置，将 Deep Fashion 数据集从 289,222 张图片缩减到了 200,000 张图片进行实验。在实验结果中，在 epoch 较高时，模型明显出现了过拟合现象。这是由于缩短后的训练集数据不充分，导致训练集无法很好地描述服装信息的分布规律，导致即使模型已经很好地拟合了训练集，但是在测试集上却无法取得非常优秀的性能。所以，在未来研究中，可以更换更强算力的设备，使用更大规模的 Deep Fashion 数据集，来让模型取得更好的性能。

8.2.2 复杂化损失函数设计

在本实验中，对于多任务模型损失采用的是加权求和的损失函数设计，并通过实验确定了每个权重的具体大小。但是，在模型中，可能不同 loss 之间的关系比线性关系更加复杂。在训练过程中，还可能需要动态的改变每个任务在 loss 中的占比权重。所以，对于多任务模型的损失，可能不能仅仅通过加权求和的方式来拟合。比如，[33]中介绍了一种加入惩罚项的拟合方式，该方式为每个任务的损失增加了基于一个当前训练梯度大小的惩罚项，该惩罚项可以做到在训练过程中动态约束每个任务的损失值。在[34]中，介绍了用将梯度用几何方式组合($L = \sum_{i=1}^n \sqrt[n]{L_i}$)来替代加权求和来替代加权求和的损失函数设计方案，并证明了这种设计方案能够更好的解决不同任务收敛速度不一样的问题。在[35]中，介绍了一种通过 loss 的均值和标准差之间的变化情况来计算不同 loss 之间的权值，并指出这种设计方案能让 loss 的变化曲线更加平滑，也能让不同任务更加均衡。所以，在后续研究中，可以尝试更加复杂的复合损失函数设计方案，并开展相

关实验来研究它们对网络性能的影响。

8.2.3 精细化数据集

虽然 Deep Fashion 数据集是现在时尚视觉领域使用最为广泛的数据集，但其实该数据集还存在优化空间。第一，Deep Fashion 数据集均是人为标注，这会导致一些错误数据标注。第二，Deep Fashion 数据集图像质量很低，这会导致某些图像细节失真，无法被模型识别。目前，已经有相关工作在改进 Deep Fashion 数据集。比如，论文[16]提出了 Deep Fashion2，它采用更高清的图像，并使用一种高准确率的 RCNN 模型来标注 Bounding Box 标签，使得 Bouding Box 更加精确。在未来的研究中，可以尝试替换数据集，来对比模型在不同数据集上的拟合性能。

8.2.4 从自动标注到自动服装分割

在本文对 Deep Fashion 数据集的预处理中，直接采用了 Deep Fashion 数据集提供的 Bounding Box 数据来提取服装边界。然而，在真正使用模型时，还需要手动对输入的图片划分 Bounding Box。如何让神经网络自动识别服装区域以提取服装边界的问题便是自动服装分割问题。目前，服装图像自动分割技术主要采用循环卷积神经网络(RCNN)。比如，论文[16]介绍了一种带掩膜的循环神经网络(Mask-RCNN)来自动提取服装区域。在未来的研究中，可以研究相关技术，实现一个服装分割模型。该模型可以用于自动提取测试图片的服装区域，并进入本文的模型进行预测，形成一个完整的自动标注系统。

8.2.5 从标注数据到智能时尚业务

本文的模型输出结果包括服装图像的类别、属性标签以及中间产物 land mark 边界点坐标。这些数据在智能时尚领域都有很高的应用价值。例如，服装类别和属性标签可以为相关推荐算法提供数据支持，也可以作为服装检索算法的索引。而 landmark 边界点则可以运用在人物识别、不同动作下的服装匹配以及智能穿搭等。所以，未来的研究也可以转向应用层面，以本文的模型作为基础，实现其他智能时尚业务，真正将自动标注技术转化为实用价值。

参考文献

- [1] 郝杰,李涛.《2018/2019 中国纺织服装行业年度报告》[J].纺织服装周刊
- [2] Lowe, David G. "Distinctive image features from scale-invariant keypoints." *International journal of computer vision* 60.2 (2004): 91-110..
- [3] Dalal, Navneet, and Bill Triggs. "Histograms of oriented gradients for human detection." *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*. Vol. 1. Ieee, 2005.
- [4] Chen, Huizhong, Andrew Gallagher, and Bernd Girod. "Describing clothing by semantic attributes." *European conference on computer vision*. Springer, Berlin, Heidelberg, 2012.
- [5] Bourdev, Lubomir, Subhransu Maji, and Jitendra Malik. "Describing people: A poselet-based approach to attribute classification." *2011 International Conference on Computer Vision*. IEEE, 2011.
- [6] Bossard, Lukas, et al. "Apparel classification with style." *Asian conference on computer vision*. Springer, Berlin, Heidelberg, 2012.
- [7] Simo-Serra, Edgar, et al. "A high performance crf model for clothes parsing." *Asian conference on computer vision*. Springer, Cham, 2014.
- [8] Liu, Si, et al. "Fashion parsing with weak color-category labels." *IEEE Transactions on Multimedia* 16.1 (2013): 253-265.
- [9] Liang, Xiaodan, et al. "Human parsing with contextualized convolutional neural network." *Proceedings of the IEEE international conference on computer vision*. 2015.
- [10] Yamaguchi, Kota, et al. "Parsing clothing in fashion photographs." *2012 IEEE Conference on Computer vision and pattern recognition*. IEEE, 2012.
- [11] Abdulnabi, Abrar H., et al. "Multi-task CNN model for attribute prediction." *IEEE Transactions on Multimedia* 17.11 (2015): 1949-1959.
- [12] Huang, Junshi, et al. "Cross-domain image retrieval with a dual attribute-aware ranking network." *Proceedings of the IEEE international conference on computer vision*. 2015.
- [13] Lin, Min, Qiang Chen, and Shuicheng Yan. "Network in network." *arXiv preprint arXiv:1312.4400* (2013).
- [14] Liu, Ziwei, et al. "Deepfashion: Powering robust clothes recognition and retrieval with rich annotations." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- [15] Gidaris, Spyros, and Nikos Komodakis. "Object detection via a multi-region and semantic segmentation-aware cnn model." *Proceedings of the IEEE international conference on computer vision*. 2015.
- [16] Ge, Yuying, et al. "Deepfashion2: A versatile benchmark for detection, pose estimation, segmentation and re-identification of clothing images." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019.
- [17] Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
- [18] Bossard, Lukas, et al. "Apparel classification with style." *Asian conference on computer vision*. Springer, Berlin, Heidelberg, 2012.
- [19] Hadi Kiapour, M., et al. "Where to buy it: Matching Street clothing photos in online shops." *Proceedings of the IEEE international conference on computer vision*. 2015.
- [20] Chen, Qiang, et al. "Deep domain adaptation for describing people based on fine-grained clothing attributes." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.
- [21] Huang, Junshi, et al. "Cross-domain image retrieval with a dual attribute-aware ranking network." *Proceedings of the IEEE international conference on computer vision*. 2015.
- [22] LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.
- [23] Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems* 25 (2012).
- [24] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

- [25] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.
- [26] Ding, Xiaohan, et al. "Repvgg: Making vgg-style convnets great again." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2021.
- [27] Huang, Gao, et al. "Densely connected convolutional networks." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.
- [28] Szegedy, Christian, et al. "Inception-v4, inception-resnet and the impact of residual connections on learning." *Thirty-first AAAI conference on artificial intelligence*. 2017.
- [29] Wang, Zhou, et al. "Image quality assessment: from error visibility to structural similarity." *IEEE transactions on image processing* 13.4 (2004): 600-612.
- [30] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).
- [31] Robbins, Herbert, and Sutton Monro. "A stochastic approximation method." *The annals of mathematical statistics* (1951): 400-407.
- [32] Deng, Jia, et al. "Imagenet: A large-scale hierarchical image database." *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009.
- [33] Kendall, Alex, Yarin Gal, and Roberto Cipolla. "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2018.
- [34] Chennupati, Sumanth, et al. "Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*. 2019.
- [35] Groenendijk, Rick, et al. "Multi-loss weighting with coefficient of variations." *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. 2021.
- [36] Lu, Yongxi, et al. "Fully-adaptive feature sharing in multi-task networks with applications in person attribute classification." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017.
- [37] Wang, Wenguan, et al. "Attentive fashion grammar network for fashion landmark detection and clothing category classification." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018.

装
订
线

謝 辞

光阴荏苒，四年的本科时光转瞬即逝，在软件学院这段难忘的时光里，我得到了太多人的关怀与帮助，在这里，我想向他们致以最真诚的谢意。

首先我要感谢我的父母以及亲人。四年本科的本科生活中，我犯过不少错误，给你们带来了不少麻烦，但是不论什么情况下，你们仍然义无反顾地支持我。在物质上，你们无条件地满足我的各种要求。没有你们，我就不可能有进入软件学院学习的机会。有时，我因为学业而感到焦虑，因人际关系而突生沮丧，也因出国事宜陷入忧虑，在这些时候你们总能站出来，无视我的冲动情绪，用你们温柔的关怀，给我一个温暖的港湾。这样无私伟大的养育之恩，我一定会永远铭记。我一定会更加努力地完成我的研究生学业，不让你们失望。

其次，我想要感谢身边的朋友，谢谢你们陪我度过了这难忘的本科生涯。在学习上，我要感谢我出色的队友们，谢谢你们能带着我，共同努力，出色地完成一个又一个的课程项目。在日常生活中，我特别感谢我的室友们，你们经常帮助解决一些生活小事，让我的校园生活快乐而顺利。在课余生活中，我要感谢我在体育部还有篮球队的伙伴们，你们教会了我太多组织活动方面的技能，是你们让我的大学生活充满了青春热血。在校外，我要特别感谢我在北京实验室的老师和朋友们，能与优秀的你们一起共事，我学到了太多太多。在这里获得的推荐信，也在我的出国申请中起到了至关重要的作用。当然，我还要感谢我在重庆的朋友，在课余时光里，我们无话不谈，你们帮我消除了太多学习上的烦恼。

再者，我要感谢软件学院老师们。是你们让我从一个青涩的高中生成长为了一位全面发展的大学生。在计算机视觉课程上，张老师精彩而细心的讲解让我学到了很多专业知识；在用户交互课程上，沈莹老师有趣的课程设置让我爱上了交互设计，为我的研究生学习指明了方向；在我的毕业设计上，刘琴老师给了我太多指导，无论是选题还是实际操作，她都耐心地解答我的疑问，给我倾注了很多关怀和鼓励。还有太多软件学院的老师，在此不能一一列举，我借此机会对您们表示深深的谢意。同时，我也要感谢我的学长学姐们，很感谢她们对我的实验内容提供了很多中肯的建议，以及无私地提供他们的实验设备。

最后谢谢各位专家老师对本论文的查阅评审，你们的中肯的点评将转化为我进步的动力。我也要感谢软件学院这个平台，在这里经历的一切都将成为我今后学习道路中的能量，激励我以更坚定的步伐向前迈进。本科生涯在此已经接近尾声，我一定会以更认真的态度、更饱满的精神状态，迎接我的研究生学习生活。



大学生论文检测系统

文本复制检测报告单(简洁)

No:ADBD2022R_20220601103250466541205501

检测时间: 2022-06-01 10:32:50

篇名: 基于深度学习的服装图像自动标注方法

作者: 李源峰 (1852448;软件学院;软件工程)

指导教师: 刘琴

检测机构: 同济大学

提交论文IP: 111.***.***.***

文件名: 1852448李源峰毕业设计——基于深度学习的服装自动标注方法.docx

检测系统: 大学生论文检测系统

检测类型: 大学生论文

检测范围: 中国学术期刊网络出版总库

中国博士学位论文全文数据库/中国优秀硕士学位论文全文数据库

中国重要会议论文全文数据库

中国重要报纸全文数据库

中国专利全文数据库

图书资源

优先出版文献库

大学生论文联合比对库

互联网资源(包含贴吧等论坛资源)

英文数据库(涵盖期刊、博硕、会议的英文数据以及德国Springer、英国Taylor&Francis 期刊数据库等)

港澳台学术文献库

互联网文档资源

源代码库

CNKI大成编客-原创作品库

机构自建比对库

时间范围: 1900-01-01至2022-06-01

检测结果

去除本人文献复制比: 0.5% 跨语言检测结果: 0%

去除引用文献复制比: 0.5% 总文字复制比: 0.5%

单篇最大文字复制比: 0.1% (基于卷积神经网络在手势数字识别中的研究分析)

重复字数: [244] 总段落数: [5]

总字数: [50515] 疑似段落数: [2]

单篇最大重复字数: [71] 前部重合字数: [184]

疑似段落最大重合字数: [184] 后部重合字数: [60]

疑似段落最小重合字数: [60]



■ 文字复制部分 0.5%

■ 无问题部分 99.5%

指标: 疑似剽窃观点 疑似剽窃文字表述 疑似整体剽窃 过度引用

相似表格: 0 相似公式: 没有数据 疑似文字的图片: 0

1.7% (184) 1.7% (184) 基于深度学习的服装图像自动标注方法 第1部分 (总10560字)0% (0) 0% (0) 基于深度学习的服装图像自动标注方法 第2部分 (总11560字)
(0) 0% (0)

0%		基于深度学习的服装图像自动标注方法 第3部分 (总11937字)
0% (0)		基于深度学习的服装图像自动标注方法 第4部分 (总10693字)
1% (60)		基于深度学习的服装图像自动标注方法 第5部分 (总5765字)

指导教师审查结果

指导教师：刘琴

审阅结果：

审阅意见：指导老师未填写审阅意见

1. 基于深度学习的服装图像自动标注方法_第1部分总字数：**10560****相似文献列表**

去除本人文献复制比：1.7%(184) 文字复制比：1.7%(184) 疑似剽窃观点 (0)

1	基于卷积神经网络在手势数字识别中的研究分析 曾祥强;刘瑞;杨鑫; - 《物联网技术》 - 2021-06-20	0.7% (71) 是否引证：否
2	服装智能搭配研究现状综述 张泽堃;张海波; - 《网络安全技术与应用》 - 2019-10-15	0.6% (65) 是否引证：否
3	120160483_李思穆_基于Mask R-CNN的前列腺TRUS图像分割方法研究 李思穆 - 《大学生论文联合比对库》 - 2020-05-08	0.4% (40) 是否引证：否

2. 基于深度学习的服装图像自动标注方法_第2部分总字数：**11560****相似文献列表**

去除本人文献复制比：0%(0) 文字复制比：0%(0) 疑似剽窃观点 (0)

3. 基于深度学习的服装图像自动标注方法_第3部分总字数：**11937****相似文献列表**

去除本人文献复制比：0%(0) 文字复制比：0%(0) 疑似剽窃观点 (0)

4. 基于深度学习的服装图像自动标注方法_第4部分总字数：**10693****相似文献列表**

去除本人文献复制比：0%(0) 文字复制比：0%(0) 疑似剽窃观点 (0)

5. 基于深度学习的服装图像自动标注方法_第5部分总字数：**5765****相似文献列表**

去除本人文献复制比：1%(60) 文字复制比：1%(60) 疑似剽窃观点 (0)

1	中小企业信用指标体系构建及评估模型的最优化 奚梦缘(导师：刘海飞) - 《南京大学硕士论文》 - 2018-05-23	0.5% (31) 是否引证：否
2	基于卷积神经网络的乳腺肿瘤良恶性分类方法研究 边坤鹏(导师：于炯) - 《新疆大学硕士论文》 - 2020-05-13	0.5% (29) 是否引证：否

说明： 1.总文字复制比：被检测论文总重合字数在总字数中所占的比例

2.去除引用文献复制比：去除系统识别为引用的文献后，计算出来的重合字数在总字数中所占的比例

3.去除本人文献复制比：去除作者本人文献后，计算出来的重合字数在总字数中所占的比例

4.单篇最大文字复制比：被检测文献与所有相似文献比对后，重合字数占总字数的比例最大的那一篇文献的文字复制比

5.指标是由系统根据《学术论文不端行为的界定标准》自动生成的

6.本报告单仅对您所选择比对资源范围内检测结果负责

 amlc@cnki.net



 check.cnki.net

<http://check.cnki.net/>