



汕 頭 大 學

SHANTOU UNIVERSITY

硕 士 学 位 论 文

题 目 基于深度学习的钢筋端面中心点检测及计数

英文题目 Automated Steel bar Counting and Center Localization
 Based on Deep Learning

姓 名 姜涛 学 号 111709016

所在学院 工学院 导师姓名 姜永权

专 业 信息与通信工程

入学日期 2017 年 9 月 答辩日期 2020 年 6 月

学位论文原创性声明

本论文是我个人在导师指导下进行的工作研究及取得的研究成果。论文中除了特别加以标注和致谢的地方外，不包含其他人或其它机构已经发表或撰写过的研究成果。对本文的研究做出贡献的个人和集体，均已在论文中以明确方式标明。本人完全意识到本声明的法律责任由本人承担。

作者签名：_____ 日期：_____年___月___日

学位论文使用授权声明

本人授权汕头大学保存本学位论文的电子和纸质文档，允许论文被查阅和借阅；学校可将本学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或其它复制手段保存和汇编论文；学校可以向国家有关部门或机构送交论文并授权其保存、借阅或上网公布本学位论文的全部或部分内容。对于保密的论文，按照保密的有关规定和程序处理。

作者签名：_____ 导师签名：_____

日期：_____年___月___日 日期：_____年___月_____

摘要

钢铁工业是许多国家最重要的基础产业之一,而工厂的自动化流水生产线对提高钢筋的生产率起着至关重要的作用。目前国内钢筋生产厂商还是采用人工对钢筋进行分离和计数的方式,现场的工人只能采取简单的防护设备对钢筋进行捆包装和计数。这种方式的缺点是工作效率低且计数的误差较大,此利用数字图像处理技术实现钢筋端面中心点检测和计数,可大幅降低他们的劳动强度,提高生产效率。基于深度学习算法,针对不同的应用场景,本文提出了两种钢筋端面中心点检测和计数算法。

第一种方法,首先用 CNN 检测钢筋端面候选中心点。在此基础上,提出了一种距离聚类(DC)算法来聚类候选中心点定位钢筋的中心。该方法适用于固定场景的钢筋端面中心点检测及计数,如相机拍摄钢筋端面的距离和角度几乎不发生变化的场景。仿真结果表明,CNN-DC 算法可以实现 99.26%的钢筋计数精度和 4.1%的中心偏移量。

第二种方法,基于卷积神经网络的密度估计对钢筋图片进行识别,并通过编码器和解码器得到钢筋的单一特征图,对预测的密度图进行积分实现钢筋端面中心点的检测和计数功能。该方法适用于复杂场景下的钢筋端面中心点检测及计数。仿真结果表明,基于卷积神经网络的密度估计算法可以实现 94.03%的钢筋计数精度和 5.61%的中心偏移量。

将本文的两种方法与传统方法对钢筋数据集的实验结果对比可以看出,本文中的两种方法在识别钢筋端面中心的偏移量,计数准确率等指标上均比传统方法的表现力更好。

关键词: 钢筋计数; 中心定位; 卷积神经网络; 聚类; 密度估计

Abstract

The steel industry is one of the most important industries in many countries. At present, the domestic steel bar factory still adopt the manual method to conduct automated center points detection and counting of steel bars, whose performances are limited by complex industrial environments. It has become an urgent problem to solve by using digital image processing technology to realize automated steel bar counting and center localization. In this paper, two methods of automated center points detection and counting of steel bars based on deep learning are proposed.

The first method firstly uses CNN to detect the candidate center points of steel bars on this basis, an effective clustering algorithm, distance clustering (DC) algorithm, is proposed to locate the centers of steel bars. The proposed CNN-DC can achieve 99.26% accuracy for steel bar counting and 4.1% center offsets for center localization on the established steel bar dataset which demonstrates that the proposed CNN-DC is very effective in dealing with steel bar center detection and counting with a relatively stable background.

The second method based on the density estimation using convolutional neural network to realize steel bar detection and counting. This method obtains the single feature map of the reinforcement through the encoder and decoder, and integrates the predicted density map to realize the detection and counting of the center points of the steel bars. The second method can achieve 94.03% accuracy for steel bar counting and 5.61% center offsets for center localization on the established steel bar dataset.

Compared with other traditional methods on steel bar data set, it can be seen that the two methods in this paper are better than the traditional methods in several performance metrics including the offset of the center of the steel bar, counting accuracy and other indicators.

Keywords: Steel bar counting; Center localization; Convolutional neural network; Density estimation; Distance clustering.

目 录

摘要.....	I
Abstract.....	II
目 录.....	III
第 1 章 绪论.....	1
1.1 研究背景及意义.....	1
1.2 国内外研究现状.....	2
1.3 研究的主要内容和章节安排.....	4
第 2 章 数字图像技术基础.....	5
2.1 数字图像处理的发展及应用.....	5
2.2 与本文相关数字图像处理技术.....	6
2.2.1 Sobel 边缘检测和 Canny 边缘检测.....	6
2.2.2 阈值分割.....	7
2.2.3 模板匹配.....	8
2.3 本文算法实现的开发平台及编程语言.....	9
第 3 章 常见的实现分类功能的卷积神经网络.....	10
3.1 LeNet 和 AlexNet.....	10
3.2 VGG.....	12
3.3 深度残差网络 ResNet.....	14
第 4 章 卷积神经网络+距离聚类(CNN-DC)实现钢筋端面中心点检测及计数.....	16
4.1 数据的准备.....	16
4.2 卷积神经网络（CNN）的搭建.....	18
4.3 聚类算法.....	21
4.3.1 KNN 和 K-means.....	21
4.3.2 本文中改进的聚类算法.....	23
4.4 CNN-DC 算法实验结果及分析.....	25
第 5 章 基于 CNN 的密度估计算法实现钢筋端面中心检测及计数.....	29
5.1 数据预处理.....	30
5.2 基于 CNN 的密度估计网络搭建.....	33
5.3 模型的评估.....	35

5.4 实验结果对比和总结.....	37
第 6 章 总结及下一步工作.....	44
6.1 论文总结.....	44
6.2 下一步工作.....	44
参考文献.....	45
攻读学位期间主要研究成果.....	49
致谢.....	50

第 1 章 绪论

1.1 研究背景及意义

钢铁工业是许多国家最重要的基础产业之一，其中钢筋是最常用的钢材产品之一，主要用于建筑施工，与人们的生活密切相关。工业生产的钢筋包括光圆钢筋、带肋钢筋、扭转钢筋等。其横截面为不规则的圆形或者方形（本文识别的钢筋的横截面为不规则的圆形）。

钢筋生产出来离开钢厂之前，需要对钢筋进行捆包装，如下图 1-1 所示。捆包装之后需要对其进行一个矩形铭牌的焊接，铭牌上有钢筋的信息，如该捆钢筋的数量、生产日期、发往的目的地等。实际工厂要求铭牌焊接的位置需要处于捆包装钢筋的中心位置。工厂工人焊接铭牌的时候会在矩形铭牌的 4 个顶点处焊接，因此需要焊接的时候需要找到钢筋端面的中心位置，这样焊接的铭牌在钢筋的运输过程中更加牢固，不易脱落。

工厂自动化流水生产线对提高钢筋的生产率起着至关重要的作用，其中自动化钢筋计数和中心定位是最关键的步骤之一。



图 1-1 钢筋的捆包装以及铭牌在钢筋端面的位置



图 1-2 人工对钢筋进行分离和计数

如图 1-2 所示，目前国内钢筋生产厂商还是采用人工对钢筋进行分离和计数的方式，现场的工人只能采取简单的防护设备对钢筋进行捆包装和计数。这种方式的缺点是工作效率低且计数的误差较大，另外工厂由于灰尘、高温等因素造成了工人恶劣的工作环境，对工人的人身安全产生了一定的隐患。随着科技的进步以及现代化工厂对生产效率的要求，传统的人工操作会被现代化的技术所替代。

综上所述，钢筋的中心点检测以及计数的准确率对钢筋生产有着重要的意义，因此利用数字图像处理技术实现钢筋端面中心点检测和计数，使用机械手臂替代人工焊接铭牌，实时的帮助一线工人实现钢筋的计数，降低他们的劳动强度和安全隐患，已经成为一个迫切解决的问题。

1.2 国内外研究现状

钢材工厂生产环境光线引起图像拍摄的模糊、生产线上钢筋的快速移动、高温下钢筋截面氧化引起的亮度变化、未完全对准导致的钢筋间的相互遮挡、钢筋截面形状的差异等都给钢筋的计数带来了巨大的难度。

20 世纪 70 年代，国外就已经开始对钢筋的自动计数的系统和方法进行了研究。最开始的方法是利用机械手段对钢筋进行分离和计数，如 kajii sadao^[1]等人设计了一款让自动分离钢筋的机械设备。近年来，利用图像处理技术和计算机视觉技术对钢筋进行自动检测和计数的研究取得了很大的进展。Ghazali^[2]等人提出了一种适用于识别

圆形钢筋和矩形钢筋算法,首先对采集到的数字图像进行预处理,然后进行边缘检测,提取钢筋边缘。对于圆形钢筋,采用霍夫变换和高斯拉普拉斯算子技术相结合的方法,而矩形钢筋则是通过一系列的形态学操作来识别。

90年代初,国内也有很多研究机构和科研人员开始研究钢筋的自动计数实现。张达等人^[3]首先通过光学系统优化来增强钢筋截面图像的对比度,然后在图像预处理阶段进行了多种图像增强方法,通过模板匹配来降低噪声,最后提出了一种新的自适应阈值分割方法,对分割图像的连通性进行分析,最终实现钢筋中心定位。

东北林业大学工程技术学院颖欣^[4]等人提出了一种自动分割钢筋图像的方法,首先使用 Sobel 边缘检测算子检测钢筋的边缘, Otsu 算法被用来分离钢筋和背景图片。在分割后的图片中采用改进的霍夫变换积累灰度值,最后通过阈值分割得到最后的钢筋的位置。

海军工程大学电气工程学院刘小虎^[5]等人提出了一种基于钢筋边缘轮廓的钢筋计数算法。首先生成基于灰度直方图的阈值选择的二值图像,提取二值图像中的钢筋边缘轮廓并将图片中的钢筋分类为单个钢筋轮廓或多个钢筋轮廓,最终标记不同轮廓的钢筋并对其求和实现钢筋的计数。

无锡中科普信息技术研发中心武阳^[6]等人提出了一种基于机器视觉的在线钢筋计数方法,该方法利用凹点匹配进行分段,采用 K 级容错计数,并通过视觉反馈自动进行多次分裂。首先,它对钢筋图像进行预处理,并使用连接区域分析获得粘连钢筋的边缘轮廓,然后扫描轮廓中的凹入区域并找到凹入点。接着利用凹点匹配对钢筋端面进行分割,分割后对单个条进行计数,通过运动估计和 K 级容错算法达到计数目的。最后提供视觉反馈,如果初步分割错误,会重新绘制分割线并再次驱动分割机制。

近年来,利用深度学习实现人群密度估计是一种计算机视觉任务,它将人群图像作为输入,输出相应的人群密度图,最后对该图进行求和,得到最终的行人数。

Zhang C^[7]等人利用设计用深卷积神经网络来解决跨场景人群计数问题,所学习的模型在描述人群场景方面比传统方法有更好的能力。模型采用具有两个相关学习目标即密度估计和全局计数的可切换训练策略。两个相关任务相互辅助,交替回归两个损失函数。此外,本文还提出了一种数据驱动的方法,从训练数据中选取样本,对训练前的模型进行调整,以适应未知的目标场景

Zhang Y^[8]等人设计了 MCNN 去识别行人数目,MCNN 包含了三列具有不同滤波器大小的卷积神经网络,三列对应于不同大小的感受野(大,中,小),使每个列的卷积神经网络对不同的图像分辨率造成的人头大小变化是自适应的,用一个 1*1 滤波器的卷积层代替了全连接层,模型的输入图像可以是任意大小的。网络直接输出一个人群密度估计图,从而实现人群计数。

Li Y^[9]等人设计了 CSRnet,作者提出的网络模型由前端网络 and 后端网络两个网络

组成。其中提取图像特征的前端网络采用了去掉全连接层的 VGG-16，后端网络则采用空洞卷积神经网络，整体的网络模型在保持分辨率的同时扩大了感受野的大小，可以生成高质量的人群分布密度图。

V Ranjan^[10]等人设计了 IC-CNN 网络(Iterative Crowd Counting)，其采取的思路是首先生成低分辨率的密度图，然后进一步的细化来生成高分辨率的密度图。该网络结构由两个 CNN 分支组成，一个分支用来生成低分辨率的密度图，另一个使用生成的低分辨率的密度图以及提取的特征图的基础上，生成高分辨率的密度图

Gao J^[11]等人简要介绍了一个用于人群计数任务的代码框架 C3F，它提供了主流数据集的预处理技巧和改进的神经网络的实验结果。该代码框架能够降低训练过程中的人力成本，促进人群计数的学术研究。本文提出的第二种算法将人群密度估计的思想，应用于钢筋端面中心点检测和计数。

1.3 研究的主要内容和章节安排

本文中的内容结构如下所示：第一章概述了研究背景及研究意义，国内外的研究现状，以及论文的主要章节安排；第二章介绍了数字图像处理的发展与应用，本论文相关的数字图像处理相关技术和本文系统软件开发软件，方便后续的相关计数描述以及实验对比；第三章介绍了近年来常见的实现分类功能的卷积神经网络如 AlexNet^[12]、VGG^[13]、ResNet^[14]等。本文针对应用场景提出了两种不同的解决方法。针对单一场景（即相机拍摄钢筋端面的距离和角度几乎不变化）且数据集较少的情况，第四章设计了第一种算法。首先用 CNN 检测钢筋端面候选中心点。在此基础上，提出了一种有效的聚类算法——距离聚类(DC)算法来聚类候选中心点定位钢筋的中心；为了处理多种场景不同背景下的钢筋图片，第五章设计了一个通过编码器和解码器得到单一特征图，对预测的密度图进行积分实现钢筋端面中心点的检测和计数的 CNN 网络以及本文所提出的两种算法与传统方法测试的结果对比。第六章对全文内容进行总结以及对下一步工作的说明；最后是致谢和本文的相关参考文献。

第2章 数字图像技术基础

2.1 数字图像处理的发展及应用

数字图像处理技术是将数字图像信号变成数字信号后,借助计算机完成的一中技术手段。数字图像处理技术包含了图像增强、图像的分割、图像压缩以及图像的降噪等各个方面的内容。相比于模拟图像处理,数字图像处理的优势在于其避免了外界的干扰因素的影响,即不会导致图像在处理的过程中产生图像失真、质量下降和图像模糊等现象的发生。

随着科学技术的进步与发展,图像作为信息传播的重要媒介,在人脸识别、无人驾驶、医学图像分析、航空航天等多个领域得到了广泛的研究,并在我们的社会和经济生活中承担着更加重要的角色,人们对图像研究的重视带动了计算机视觉领域的蓬勃发展。

(1) 人脸识别方面。人脸识别技术是指对图像中的内容进行分析和比较的计算机技术。基于人的脸部特征,对输入的人脸图片进行分析,返回人脸的位置、大小等信息,依据这些信息进一步提取其中的身份特征,与已知的人脸对比从而识别每个人脸的身份。涉及到的图像处理技术包括几何特征的人脸识别^[15]、基于神经网络的人脸识别^[16]等。

(2) 无人驾驶方面。无人驾驶主要依靠车内的计算机系统来实现,需要计算机视觉、导航定位、通信等多个学科技能互相协调完成。关键技术则包括环境感知、路径规划、决策控制等,其中环境感知包括对周围环境以及车身姿态的感知,数字图像处理则在其中发挥了重要的作用。

(3) 医学影像分析方面。数字图像处理技术在医学影像上的应用十分广泛,近年来快速发展的硬件技术让医学图片的成像更加清晰,另外通过数字图像处理技术可以对医学影像进行重建、分析与处理,帮助医务人员诊断疾病等。

(4) 航空航天方面。航空航天过程中,通过各种设备会对地球地貌产生各种图片,研究人员通过借助数字图像处理技术对图片进行分析,实现地面资源的调查、气象预报以及城市规划等。

2.2 与本文相关数字图像处理技术

2.2.1 Sobel 边缘检测和 Canny 边缘检测

图像灰度发生突变通常反映了图像属性的变化，边缘检测技术是数字图像特征提取的最常用的方法之一，其目的是将数字图像中亮度变化明显的像素点进行标记。衡量图像的边缘信息一般有两个要素组成，分别为边缘信息的方向和大小。一般情况下沿着边缘走向的像素值变化比较平缓；而沿着垂直于边缘的走向，像素值则变化得比较大。根据这一变化特点，数字图像处理技术通常会采用一阶导数和二阶导数来检测边缘。边缘检测能够大幅度的减少数字图像的数据，删除了一些不想关的信息的同时保留了图像的重要属性等。常用的边缘检测有 Sobel^[17]边缘检测算子、拉普拉斯^[18]边缘检测算子和 Canny^[19]边缘检测算子等。

Sobel 边缘检测算子分别包含了横向和纵向两组的 3×3 的矩阵模板，将该矩阵模板与图像卷积计算即可得到对应的亮度差分近似值。如下公式 2-1 和公式 2-2 所示， G_x 和 G_y 分别为横向和纵向的灰度偏导的近似值。

$$G_x = \begin{pmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{pmatrix} * A \quad (2-1)$$

$$G_y = \begin{pmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix} * A \quad (2-2)$$

通过下面的公式 2-3 可以计算出梯度的估计值。

$$G = \sqrt{G_x^2 + G_y^2} \quad (2-3)$$

Sobel 边缘检测算子定义一个阈值 G_m ，如果某一点计算出来的梯度值 $G > G_m$ ，则认为该点是一个边界值，这样便可得到边缘检测的图像。Sobel 边缘检测算子根据每一个像素点的邻点灰度加权，可以对图像中的噪声具有平滑作用且提供较为精确的边缘方向信息。

Canny 边缘检测算法是 John F. Canny 于 1986 年开发出来的一个多级边缘检测算法^[46]。相比于 Sobel 算子，Canny 算子更优异。Sobel 算子没有充分利用边缘的梯度

方向且最后得到的二值图只是简单的利用单阈值进行处理。Canny 算子则基于这两点做了改进，如基于边缘梯度方向的非极大值抑制，使用双阈值的滞后阈值处理等。

Canny 边缘检测实现的步骤为：

(1) 高斯平滑滤波，滤波是为了平滑图像中的噪声。如公式 2-4 所示，若图像中的一个 3×3 的窗口为 A，经过高斯滤波后的像素点的亮度值为 v。

$$V = H * A = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} * \begin{bmatrix} a & b & c \\ d & e & f \\ h & i & g \end{bmatrix} = \text{sum} \left(\begin{bmatrix} a \times h_{11} & b \times h_{12} & c \times h_{13} \\ d \times h_{21} & e \times h_{22} & f \times h_{23} \\ h \times h_{31} & i \times h_{32} & g \times h_{33} \end{bmatrix} \right) \quad (2-4)$$

(2) 计算每个像素点的梯度和方向。Canny 算子使用 4 个模板矩阵算子来检测图像中的水平、垂直以及对象边缘。

(3) 非极大值抑制。Canny 算子会抑制梯度不大的像素点，只保留最大的梯度从而达到边缘的最小化。该步骤需要借助梯度以及梯度的方向来比较某个像素点的 8 领域与中心像素点的大小。

(4) 双阈值法被用来检测和连接边缘。经过第 (3) 步的非极大值抑制后的边缘点中，梯度值大于 T1 的称为强边缘，梯度值小于 T1 大于 T2 的称为弱边缘，其他的则不属于边缘。

Canny 边缘检测算子的参数选择：Canny 算法包含许多可以调整的参数，它们将影响到 Canny 算法的计算的时间与实效^{[47][48]}。

(1) 高斯滤波器的大小选取：第一步所有的平滑滤波器尺寸的大小将会直接影响 Canny 算法的结果。如较小的滤波器产生的模糊效果也较少，这样就可以检测较小、变化明显的细线。较大的滤波器产生的模糊效果也较多，这样带来的结果就是对于检测较大、平滑的边缘更加有效。

(2) 双阈值的大小选取：使用两个阈值比使用一个阈值更加灵活，但是它还是有阈值存在的共性问题。如设置的阈值过高，可能会漏掉重要信息；阈值过低，将会把枝节信息看得很重要，实际实验时很难找到一个适用于所有图像的通用阈值。

Canny 边缘检测的效果显著，相比于 Sobel 算子和拉普拉斯算子极大的抑制了噪声所带来的影响，便于后续图像的处理。

2.2.2 阈值分割

阈值分割技术是图像分割方法中经典方法，是一种广泛应用的分割技术。阈值分

割技术^[49]利用图像中要提取的目标区域与其背景在灰度特性上的差异,把待分割图像看作具有不同灰度级的集合,通过选取一个合理的阈值 thd 来区分图像中的某个像素点属于目标区域还是背景区域,从而实现目标区域和背景图片的分离。阈值分割利用封闭且连通的边界定义不交叠的区域,对目标与背景有较强对比的图像可以得到较好的分割效果。但是阈值的选取决定了分割的效果。目前主要有以下几种阈值的选择方法^[50]:

(1) 人工调参的方法。根据实验者对需要处理图片的像素分析,人工选择出阈值区间,通过实验结果的对比选择出最佳的阈值。这种方法虽然可以实现图像的分割,但是效率较低。当样本数据较多时这种方法不适用。

(2) 利用直方图选取阈值。利用直方图进行分析,根据直方图的波峰和波谷之间的关系,选择一个最佳的阈值。这种方法适用于所处理的图片中只存在一个目标区域和一个背景区域,且两者对比较为明显时候的情形。

(3) 最大类间方差法(OTSU)^[20]。OTSU 算法是一种使用最大类别间方差自动确定阈值的方法。算法根据图像的灰度特性将图像分为前景区域和背景区域。OTSU 算法中用于测量差异的标准是最大类间差异,即如果前景区域和背景区域之间的类间差异较大,则意味着构成图像的两个部分之间的差异较大。当目标区域的一部分被错误地划分为背景或背景的一部分区域被错误地划分为目标区域时,将会导致二者差异变小。当两个区域之间的差异应最大此时的阈值为最佳阈值。

(4) 自适应阈值^[21]。对于一些光照不均匀的图像,全局阈值分割的方式会损失很多细节信息,导致分割的效果不理想。自适应阈值法的思想是根据图像不同区域亮度的分布,计算局部阈值,因此对于一幅图像中的不同区域能够自适应计算不同的阈值。具体的可以通过计算图像中局部的均值、中值、高斯加权平均(高斯滤波)来确定阈值。

阈值分割的优点是计算简单、运算效率较高、速度快。图像的阈值分割技术已被应用于很多的领域^[51],例如,在红外技术应用中,红外检测中红外热图像的分割^[22];在遥感应用中,合成孔径雷达图像中目标的分割^[23]等;在医学应用中,血液细胞图像的分割^[24]、磁共振图像的分割^[25];在农业工程应用中,水果品质检测过程中水果图像与背景的分割^[26]。

2.2.3 模板匹配

模板匹配^[27]是一种最基本的模式识别方法之一,是数字图像处理中最常用的匹配方法。模板匹配是在整个图像区域找到与给定子图像匹配的区域,在待测图像上从左到右,从上到下计算模板图像与给定子图像的匹配程度,匹配程度越大则说明二者相

同的可能性越大。其中计算匹配度的大小方法有平方差匹配法、归一化平方差匹配法、归一化相关系数匹配法等。假设得到的图像矩阵为 $R(x, y)$ ，模板子图像矩阵为 $T(x', y')$ ，待检测图像矩阵为 $I(x, y)$ 。平方差匹配法则表示为：

$$R(x, y) = \sum_{x', y'} (T(x', y') - I(x + x', y + y'))^2 \quad (2-5)$$

上式表示为模板子图像的像素值与源图像像素值的差值的平方和，若该值越接近于 0 说明匹配的程度越高。模板匹配算法具有自身的局限性^[52]，主要表现在它只能进行平行移动，若子图像模板对应原图像中的匹配目标发生旋转或大小变化，该算法无法很好的找到所匹配的目标区域。

2.3 本文算法实现的开发平台及编程语言

在本文中 CNN-DC 算法的实现基于 Tensorflow 平台实现的，而基于密度估计的 CNN 则是依赖于 Pytorch 平台，编程的语言为 Python 实现。计算机硬件的 CPU 是在 Intel Xeon E5-2690，使用的 GPU 是 TITAN Xp。

第3章 常见的实现分类功能的卷积神经网络

图像的分类是目前计算机视觉领域最为常见的基础任务之一，是目标检测、图像分割、人脸识别等视觉任务的基础。图像的分类可以看成是一个模式分类的问题，根据一张图片的语义信息对不同类别的图像进行区分，所解决的问题是以最小的分类误差将不同的图像划分到不同的类别。

图像的分类可分为两大类，分别是单标签的图像分类和多标签的图像分类，单标签的图像分类是指一张图片对应一个类别标签。针对于单标签的图像分类又可分为二分类和多类别分类。现实生活中的图片一般包含了多个类别的物体，一张图对应了多个标签，即称为图像多标签分类问题。本文中的卷积神经网络部分主要用来解决单标签的分类。

针对图像的分类问题，传统机器学习算法的解决思路为：特征提取+分类器。其中特征提取阶段采用 HOG 特征^[28]、haar 特征^[29]等对图像进行特征提取，当选择了能够达到要求的准确率的特征后，使用 Adaboost^[30]，SVM^[31]等分类方法训练出一个分类模型。传统的机器学习算法需要组合的解决方案方案，参数众多且难以优化，无法实现端到端的训练。深度学习中的卷积神经网络在这一方面解决了传统的机器学习存在的问题，即将特征提取和分类器整合到一整套框架中，极大的避免了图像前期复杂的预处理过程，实现了端到端的训练且大幅度提高了训练的速度和分类的准确率。

3.1 LeNet 和 AlexNet

卷积神经网络由若干数量的神经元组成，单个神经元只对图像特定部分进行特征提取，如横向或者纵向的边缘。感受野是卷积神经网络的核心，卷积神经网络的卷积核则是感受野的结构表现。通过激活函数使模型获得非线性的拟合能力的同时增加了模型的表达能力。卷积神经网络一般包含卷积层（convolution layer）、池化层（pooling layer）、全连接层（fully-connected layer）等。对于图片中的每一个特征采取局部感知的策略，在更高层次的网络对局部特征进行组合操作，从而得到全局的信息，最终发掘出图片的局部关联性质以及空间的性质，卷积层的作用则在此。池化层又称为下采样层或者欠采样层，主要有最大池化层（Max Pooling）和平均池化层（Average Pooling）。其用于图像特征的降维，减少参数的数量和提高模型的鲁棒性。全连接层又称为输出层，在经过卷积层和池化层后，原始图像的数据会被映射到一个特征空间，全连接层则降其分布式的特征重新映射到另外一个样本标记空间。

1998 年 Yan LeCun^[32]等人将卷积神经网络 LeNet5 应用到图像分类任务上，成功应用于手写体识别。LeNet5^[32]是最早的卷积网络之一，是基于梯度的反向传播算法对网络进行有监督的训练。下图 3-1 所示为 LeNet5 的网络结构。

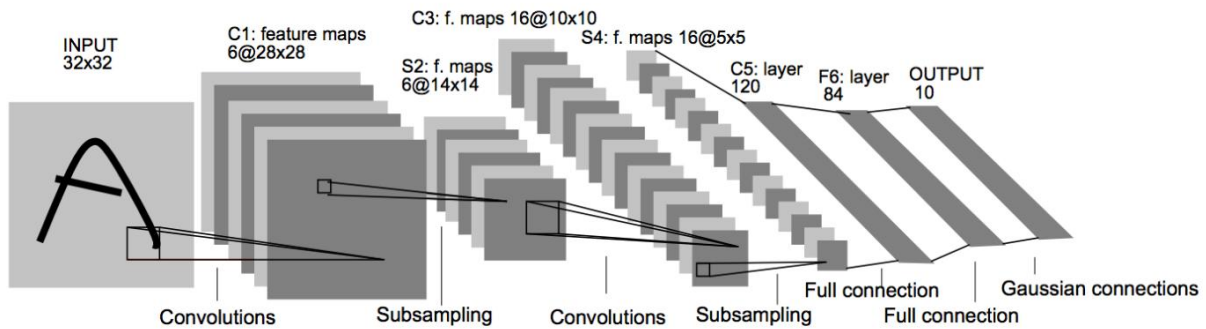


图 3-1 LeNet5 网络结构

LeNet5 由 7 层网络层组成，分别为输入层、卷积层、池化层、卷积层、池化层、卷积层、全连接层，输出层。上图的网络结构中输入的原始图像大小是 32×32 像素，卷积层用 C_i 表示，采样层用 S_i 表示，全连接层用 F_i 表示。为了避免模型的训练中出现过拟合的情况，LeNet5 采用了 Dropout 操作，即在模型的训练过程中随机让一些隐藏层节点的权重归 0，这样从另外一个角度提升了网络的适用性和鲁棒性。除此之外，LeNet5 采用了 Batch Normalization (BN) 算法，对网络的每一层网络的特征进行归一化处理，使得每一层的特征分配较为稳定。通过 BN 算法可以使用较大的学习率来训练网络，加速了模型的收敛速度。另外 BN 算法也会小幅度提高模型的分类精度。

2012 年 Hinton 和他的学生 Alex Krizhevsky 设计了 AlexNet 网络在当年的 LSVRC-2010 比赛中获得了冠军。AlexNet 在测试集 TOP-1 和 TOP-5 的识别错误率分别为 37.5% 和 17%。AlexNet 也正式引发了卷积神经网络在各个领域应用的热潮，它为后续的 CNN 网络几乎都定下了基调。

AlexNet 的网络结构和 LeNet 的网络结构类似，均由卷积层、池化层和全连接层组成。AlexNet 网络 5 层卷积层和 3 层全连接层组成。其输入为 224*224*3 的图片，通过最后一层的全连接层的 softmax 输出 1000 类的分类结果。其中 3 层的全连接层带来了大量的参数（包括 6000 万个参数和 65 万个神经元），下图 3-2 为其网络结构。

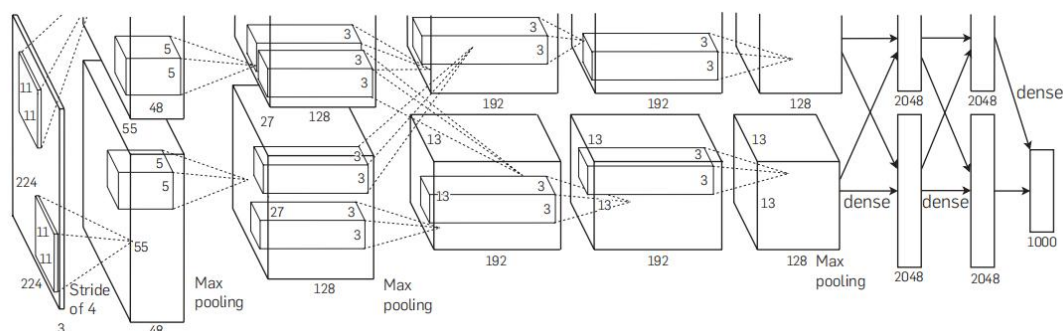


图 3-2 AlexNet 网络结构

AlexNet 网络在 LeNet5 的基础上将激活函数由 sigmoid 改为 relu 函数。随着网络层数的加深, sigmoid 激活函数在反向求导的时候会在端值趋于饱和, 造成训练速度减慢和梯度弥散的情况发生。relu 激活函数的特点是当输入信号小于 0 时, 输出均为 0, 当输入信号大于 0 的情况下, 输出等于输入。该机制可以有效防止梯度弥散的情况发生。除此之外, 可以使网络训练更快, 增加网络的非线性和稀疏性。

AlexNet 还增加了局部相应归一化 (Local Response Normalization) 操作。对网络层的某一局部神经元创建了竞争的机制, 使得其中对输入信号响应大的值输出一个更大的信号, 反之抑制反馈较小的神经元。

另外, AlexNet 使用了 Dropout 操作, 如果 Dropout 层有 n 个神经元, 那么会形成 2^n 个不同的子结构。这些不同的子结构使用数据集并行地带权重训练, 在预测时, 相当于集成这些模型并取均值。这种结构化的模型有利于避免过拟合。Dropout 另外的作用是: 由于神经元是随机选择的, 所以可以减少神经元之间的相互依赖, 从而确保提取出相互独立的重要特征。

3.2 VGG

2014 年, 来自牛津大学的 Visual Geometry Group 小组提出的 VGG^[13]卷积神经网络在 ILSVRC 竞赛上获得了分类任务第二名、定位任务第一名的成绩。在比赛中, 该小组主要研究了卷积神经网络的网络层深度对其在大规模图像识别中识别的准确度的影响, 提出的 VGG 网络为 VGG-16 和 VGG-19, 两个网络只在网络层数不同, 分别为 16 层和 19 层。下图 3-3 为 2014 年该小组提出的 VGG 网络结构示意图。

ConvNet Configuration					
A	A-LRN	B	C	D	E
11 weight layers	11 weight layers	13 weight layers	16 weight layers	16 weight layers	19 weight layers
input (224×224 RGB image)					
conv3-64	conv3-64 LRN	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64	conv3-64 conv3-64
maxpool					
conv3-128	conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128	conv3-128 conv3-128
maxpool					
conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256	conv3-256 conv3-256 conv1-256	conv3-256 conv3-256 conv3-256	conv3-256 conv3-256 conv3-256 conv3-256
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512	conv3-512 conv3-512 conv1-512	conv3-512 conv3-512 conv3-512	conv3-512 conv3-512 conv3-512 conv3-512
maxpool					
FC-4096					
FC-4096					
FC-1000					
soft-max					

图 3-3 VGG 的网络结构

从上图 3-3 可以看出 VGG 网络由卷积层、池化层和全连接层组成。在论文中分别使用了 A、A-LRN、B、C、D、E 这 6 种网络结构进行测试，这 6 种网络结构相似，都是由 5 层卷积层、3 层全连接层组成，其中区别在于每个卷积层的子层数量不同，从 A 至 E 依次增加，总的网络深度从 11 层到 19 层，表格中的卷积层参数表示为 conv（感受野大小）-（通道），例如 conv3-128，表示使用 3x3 的卷积核，通道数为 128。其中，网络结构 D 就是 VGG-16，网络结构 E 就是 VGG-19。VGG 网络相对于 AlexNet 网络有以下改进：

（1）VGG 网络使用了 3 个 3x3 卷积核来代替 7x7 卷积核，使用了 2 个 3x3 卷积核来代替 5x5 卷积核。由于多层非线性层的叠加不仅可以提升网络的深度，还增加特征的非线性组合，保证了网络学习更复杂的特征。因此 VGG 采用了多个小的卷积核的堆积代替尺寸大的卷积核。除此之外，多个小卷积核的堆积比尺寸大的卷积核所带来的参数量更少。

（2）VGG 网络的池化层全部采用了最大池化的策略，步长为 2，尺寸为 2x2。

(3) VGG 网络的搭建过程中使用了同样大小的卷积核尺寸 (3x3)，使其结构非常简洁。

(4) VGG 网络同时也验证了通过适当加深网络层数，可以提升网络的表现能力。VGG 网络对图片的特征提取能力很强，另外拓展性也很强，迁移到其它样本数据上的泛化性也非常好。

VGG 网络同样也带来了一些缺点，随着网络层数的加深且同样使用了 3 个全连接层，由此带来的参数耗费了计算机的计算资源，不利于 VGG 模型部署到移动端，以及实时的处理数据等。

3.3 深度残差网络 ResNet

随着深度卷积网络的网络层数加深，模型的识别准确率会随之上升，但是当模型的层数加到一定的程度后，所达到的效果不升反降，模型的性能趋于饱和甚至发生了退化。卷积神经网络的深度提升不能通过层与层的简单堆叠来实现。因为网络的训练是通过对梯度的反向求导实现，当梯度很小而网络层数加深的时候容易出现梯度消失的情况，使得深层网络的训练很难进行。另外随着网络层数的加深，由于非线性激活函数 Relu 的存在，每次网络的输入到输出的过程都是不可逆的，所以会因此产生额外的信息损失，很难从输出反推回完整的输入。

为了解决该问题，2015 年何凯明^[14]人提出一种深度残差网络 ResNet，在该网络中提出了一种 Residual 模块来帮助网络实现恒等映射。如下图 3-4 所示，Residual 模块能让模型的内部结构拥有恒等映射的能力，即通过学习残差表征函数而不是直接学习目标表征，以保证在堆叠网络的过程中，网络至少不会因为继续堆叠而产生退化。

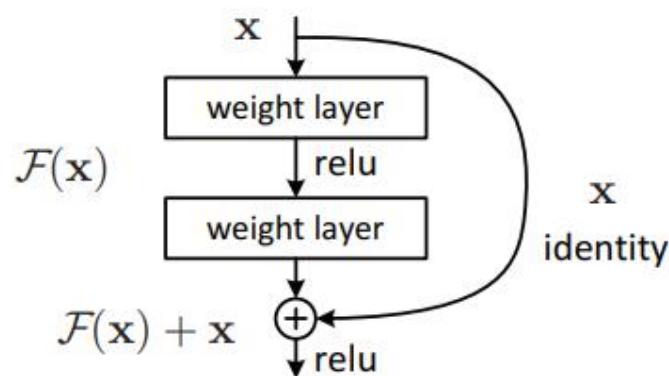


图 3-4 为残差模块结构

如上图 3-4 所示，Residual 模块将上一个浅层网络的输出加给深层的输出，

$F(X) = H(X) - X$ ，其中 X 为上一层浅层网络的输出， $H(X)$ 为深层网络的输出。 $F(X)$ 为两层网络中间的变换函数。

若将输入设为 X ，将某一有参网络层设为 H ，那么以 X 为输入的此层的输出将为 $H(X)$ 。卷积神经网络如 AlexNet 和 VGG 会直接通过训练学习出参数函数 H 的表达，从而直接学习从 X 到 $H(X)$ 的映射。

残差卷积神经网络 ResNet 则是致力于使用多个有参网络层来学习输入、输出之间的参差即 $H(X) - X$ 即学习 X 到 $[(H(X) - X) + X]$ 的映射。其中 X 为直接的 identity mapping，即 identity mapping 函数可以使得处理后的输入 X 与输出 Y 的通道数目相同。而 $H(X) - X$ 则为有参网络层要学习的输入输出间残差。

Residual 模块会明显减小模块中参数的值从而让网络中的参数对反向传导的损失值有更敏感的响应能力，这样的操作可以让网络的参数减小，相对而言也减小了信息的损失，也对数据产生了正则化作用，增加模型的泛化能力。

第 4 章 卷积神经网络+距离聚类(CNN-DC)实现钢筋端面中心点检测及计数

工厂里钢筋生产出来以后,为了实现对钢筋端面的中心点以的检测及对钢筋的自动计数,本文提出了采用了卷积神经网络(CNN)和距离聚类(Distance Clustering)相结合的算法(CNN-DC)。本文中使用了卷积神经网络来检测候选中心点。神经网络受到生物过程的启发,可用于处理各种高维数据,如图像、视频、语音信号和文本字符。在处理上述数据时,由于全连接层具有较大的特征空间,其应用有时比较繁琐。因此,卷积神经网络作为一种特殊类型的神经网络,具有空间布局、稀疏交互、参数共享等重要特性。下图所示为本文 CNN-DC 算法的流程框架。

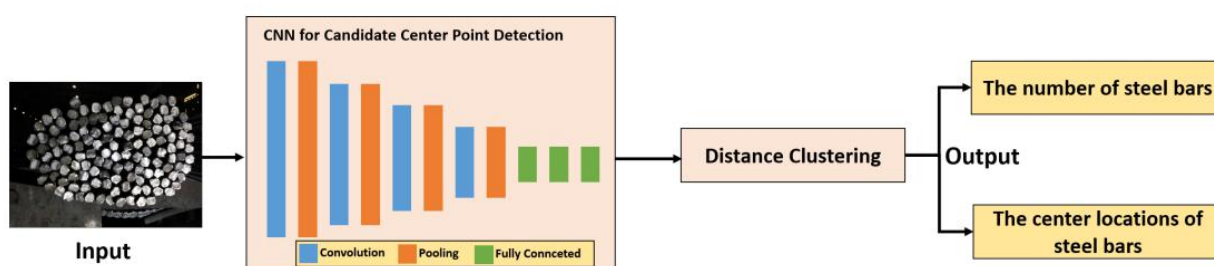


图 4-1 CNN-DC 算法处理流程框架

如上图 4-1 所示,本文所提出的 CNN-DC 算法可以看作是一种用于自动钢筋计数和中心定位的两阶段的算法。首先第一阶段利用深度卷积神经网络得到钢筋的候选中心点,接着在第一阶段的基础上提出一种有效的聚类算法——距离聚类来聚类候选中心点。算法输入的是一张灰度图,输出的是灰度图中每根钢筋端面的中心点和钢筋图片中所有钢筋的计数结果。

4.1 数据的准备

如下图所示,本文中钢筋的数据集由 10 张图像组成。这些图像是由一家钢筋工厂的工业相机获得的。每张图片分辨率高,大小为 1440×1080 。此外,每个图像由 RGB 3 个通道组成,每个通道有 8 位。将 10 幅图像分为训练集和测试集,训练集包含 4 幅图像,测试集包含 6 幅图像。从训练集中提取的 99195 个 Patch 用于训练网络。



图 4-2 原始数据集（图中红点为人工标记的 label）

Patch 的提取方法：在上图中的原始数据集中，采用滑窗的方法在钢筋端面的原始图像中遍历，窗口的大小为 71×71 （钢筋的直径），若在人工标记的 **label** 处于滑动窗口中的中心（本文规定为窗口中心 7×7 的矩形），则该窗口中的图像归为正样本即该 **Patch** 标记为 1。不满足该条件的窗口均为负样本，**Patch** 标记为 0。本文中对数据集红的 4 幅钢筋的图像进行了滑动窗口选取样本，最终我们获得了 26468 个正样本，6194332 个负样本。这样导致了我们的训练集中正负样本数据不平衡，因此我们在后续的网络训练中，首先选择 **Patch** 为 1 的数据输入卷积神经网络中，然后根据正负样本的比例（本文设置为 1:3），随机选取标记为 0 的 **Patch** 输入网络。最终我们选取了使用带有标签 1 的 26468 个正样本和带有标签 0 的 72727 个负样本来训练网络。下图 4-3 为滑窗提取 **Patch** 的示意图。

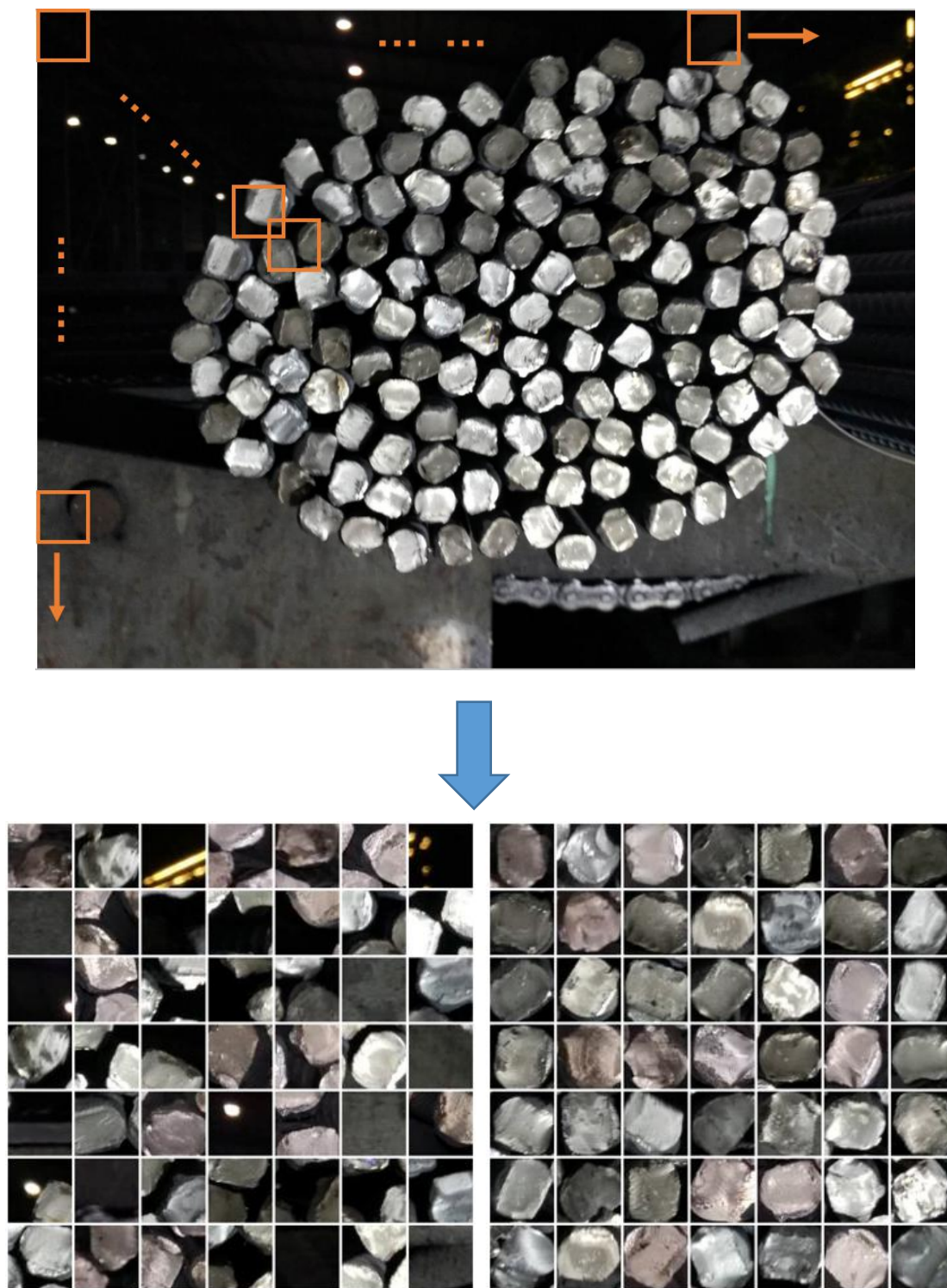


图 4-3 滑窗提取训练 CNN 的正负样本过程示意图

4.2 卷积神经网络（CNN）的搭建

卷积神经网络的输入称为张量，它由一个多维数组组成。其核心组件是卷积和池化层。卷积层将输入张量与一组卷积核进行卷积，从而生成输出张量。卷积运算是卷

积核中的各参数与图像矩阵相应位置的数值相乘后在求和，卷积核以一定的步长在输入的张量上移动。对于每一个卷积核均生成一个特征图。每一层卷积都能产生一组映射，经过多层卷积后整体构成一个复杂的函数，叠加在一起后一起生成输出张量。CNN 的训练过程可看成是在学习每个局部映射所需的权重或函数拟合的过程。

本文的卷积神经网络由 4 个卷积层 3 个全连接层完成。本文使用的 CNN 的详细信息见下图 4-4 所示。

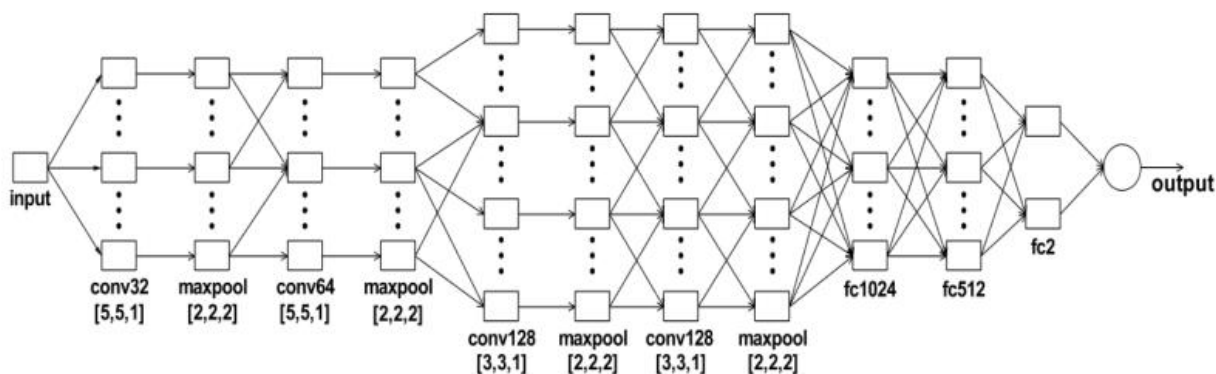


图 4-4 本文使用卷积神经网络的结构

卷积神经网络的输入为灰度图片，下图 4-5 详细的描述了 CNN 的每一层的参数：

从输入层到最后输出层每一层卷积核的数目如下所示：

输入→卷积层（32）→池化层→卷积层（32）→池化层→卷积层（32）
→池化层→全连接层（128）→全连接层（2）

其中括号中的数值为该层的卷积核的数量。

从输入层到最后输出层每一层卷积核的尺寸如下所示：

[3x3x1] [2x2x1] [3x3x1] [2x2x1] [3x3x1] [2x2x1]

其中括号的数值为卷积核的尺寸和步长，如 3×3×1 意为卷积核的尺寸大小为 3×3，步长为 1。

图 4-5 卷积神经网络中每一层的参数说明

(1) 将输入的图片转为一个多维数组的张量 $H_i \times W_i \times D_i$ （本文为 $71 \times 71 \times 3$ ），设置四个参数:K（卷积核的个数，本文在每个卷积层的个数分别为 32, 64, 128, 128），F（卷积核的空间维数，本文为 3），S（卷积的步长）和 P（图像边缘填充 0 的个数）。

(2) 经过上述卷积后会产生大小为 $H_o \times W_o \times D_o$ 的张量。其中

$$H_o = (H_i - F + 2P) / S + 1 \quad (4-1)$$

$$W_o = (W_i - F + 2P) / S + 1, D_o = K \quad (4-2)$$

(3) 本文中卷积核的参数总共有 $(F \times F \times D_i) \times K$ ，每一个卷积核的参数为 $(F \times F \times D_i)$ 。

(4) 为了避免过拟合，本文使用池化层用于减少训练参数的数量，池化的策略为最大池化层，池化层采用的是卷积核是 2×2 且步长为 2。

本文中卷积神经网络最后一层得到每个类别的得分，通过 softmax 函数获得一个准确的概率输出。在二分类的情况下，卷积神经网络模型最后需要预测的结果只有两种情况，对于每个类别我们的预测得到的概率为 p 和 q 。此时表达式为本文中的卷积神经网络采用的损失函数为交叉熵损失函数，能够衡量同一个随机变量中的两个不同概率分布的差异程度，表示为真实概率分布与预测概率分布之间的差异。交叉熵损失函数的值越小，模型预测效果就越好。交叉熵损失函数的表达式为：

$$H(p, q) = - \sum_{i=1}^n p(x_i) \log(q(x_i)) \quad (4-3)$$

其中， $p(x)$ 表示真实概率分布， $q(x)$ 表示预测概率分布。

卷积神经网络采用梯度下降法在每一次迭代中只使用一部分训练样本 (X_i, Y_i) 进行权值参数的学习和更新，每一代的权值参数可表示为：

$$\omega^{t+1} = \omega^t - \alpha \frac{\alpha}{\alpha \omega^t} J(\omega; x_i, y_i) \quad (4-4)$$

其中， t 表示迭代的次数，取值范围为 $[3000, +\infty]$ ； α 表示学习速率，取值范围为： $[0.0003, 0.01]$ 。

训练好本文中深度卷积神经网络后，得到一个用于识别钢筋端面的分类网络模型。将训练好的模型用来测试的钢筋端面原始图像，按照固定尺寸（ 71×71 ）进行遍历，并将每一次遍历后的结果输送到训练好的所述卷积神经网络进行测试和识别，在识别出的钢筋端面上标记红点，这样得到一个钢筋端面的所有候选点，得到的识别效果图如图 4-6 所示：

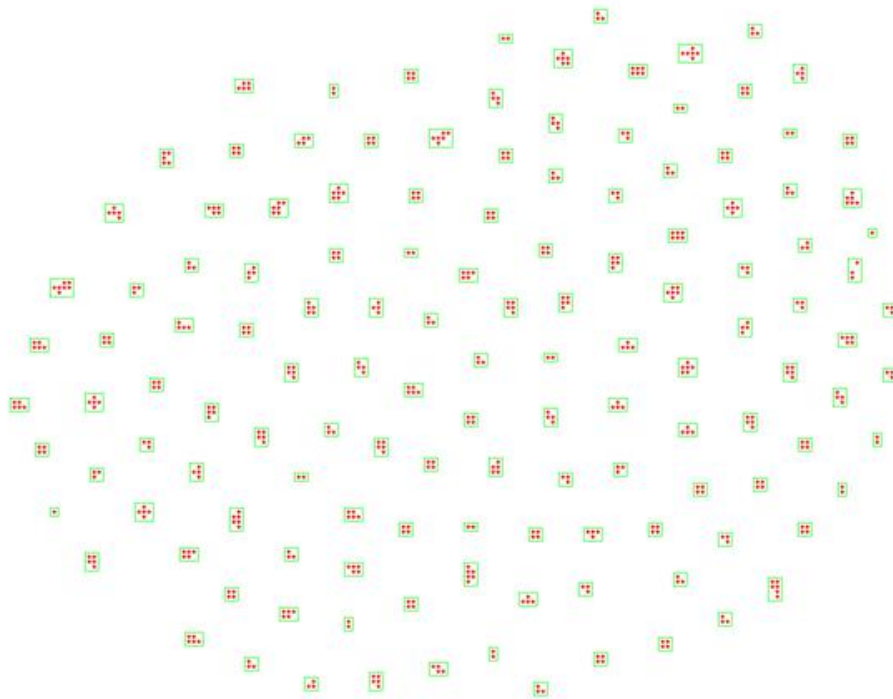


图 4-6 为本文中的分类模型识别图片中每个钢筋端面得到的候选红色点

由于需要识别的是每根钢筋中心点的位置，因此我们需要将每根钢筋得到的候选点进行聚类得到最终的中心点。

4.3 聚类算法

聚类是一种应用于分类问题的统计分析方法，同时聚类也是一中无监督学习的方法，将物理对象或者抽象对象根据规则分成多个类，每个类中的数据具有相似的属性或者特征。聚类方法包括：基于层次划分的聚类^[33]；基于连接度的聚类^[34,35]、基于中心体的聚类^[36,37]、基于密度的聚类^[38,39]和基于网格的聚类^[40]。本文提出了一种有效的聚类算法——距离聚类(Distance Cluster)算法，对从 CNN 获得的候选中心点进行聚类。

4.3.1 KNN 和 K-means

最初的分类器是将全部的训练数据所对应的类别记录下来，当训练对象与测试对象的属性匹配的时候便将两个对象分为同一类。该分类器存在的问题就是一个测试对象同时与多个训练对象匹配即一个对象被同时分到了多个类。而 KNN^[41]算法从某种程度上解决了该问题，KNN 算法又称 K 近邻算法，是通过对不同特征值之间的距离

进行测量从而进行分类，算法可以使用欧式距离、曼哈顿距离（用于计算多个维度的距离）来作为各个对象之间的非相似指标，避免了对对象之间的匹配问题。另外 KNN 算法是不是对单一的对象类别决策而是通过依据 K 个对象中占优的类别进行决策。

KNN 算法中 K 值的选取非常重要，如果 K 值的选取过小，当数据有噪声成分存在的时候，预测的类别就会产生较大的偏差，整个模型就会变得很复杂，容易出现过拟合；如果 K 值的选取过大，即用较大范围的领域中的训练数据进行预测，学习的近似误差较大且整个模型就会变得简单。因此 KNN 算法非常依赖于 K 值的选取。

KNN 算法流程为^[53]：

- （1）计算待测试对象与每个训练对象之间的欧式距离
- （2）对所有计算出来的欧式距离按照递增的顺序进行排列
- （3）取出上述排列顺序中距离最小的 K 个点
- （4）确定前 K 个点所属于类别出现的频率
- （5）返回前 K 个点出现频率最高的类别作为 KNN 算法对测试对象的预测类别。

KNN 算法对于随机分布的数据集分类效果较差，对于类内间距小，类间间距大的数据集分类效果好。当训练的数据样本很大时，需要大量的存储空间，而且需要计算待测样本和训练数据集中所有样本的距离，所以 KNN 算法非常耗时。

K-Means^[42]称 K 均值聚类，给定一个数据点集合和需要的聚类数目 K（K 值需要用户提前指定），K-means 根据距离函数将数据划分到 K 个类中。K-means 算法的流程为：

- （1）选择簇的数目 K
- （2）从测试数据中选择 K 个随机点作为质心
- （3）将所有数据分配到某一个质心距离最近的簇
- （4）重新计算新形成的簇的质心
- （5）重复第（3）（4）
- （6）直到新形成的簇的质心保持不变或者数据点保留在同一个簇中。

同样的，K-Means 算法中 K 值的选取也非常重要，在无监督聚类任务上，由于并不知道数据集有多少类别，所以很难确定 K 的取值，最优的 K 值应该与样本的数据结构相吻合，而最优的 K 值选择是非常困难的。K-Means 算法是局部最优的，容易受到初始质心和异常数据的影响，若初始质心的选取不是最佳的则最终可能达到次优的聚类结果。另外 K-Means 由于采用欧氏距离来衡量样本之间相似度，所以得到的聚簇都是凸的，就不能解决“S 型”数据分布的聚类，这就使得 K-Means 的应用范围受限，难以发现数据集中一些非凸的性质。

4.3.2 本文中改进的聚类算法

本文除了要识别钢筋端面的中心点外，还需要对图片中钢筋的数量进行计数功能，因此无法使用 KNN 和 K-means 的聚类算法。算法的输入是输入：候选中心点的位置和距离阈值 thd ，输出是一组集群，包含了每个集群的中心和集群的数量。如下图 4-7 所示为距离聚类算法的实现过程。

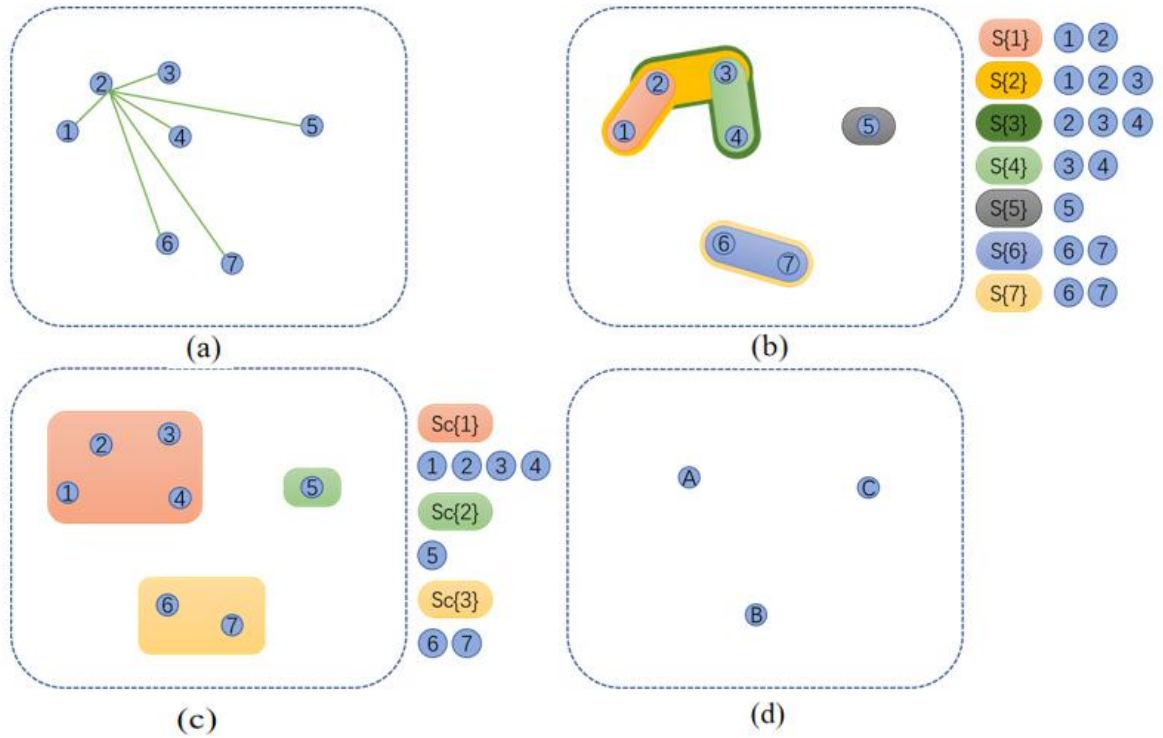


图 4-7 距离聚类算法的实现过程示意图

具体的实现步骤为：

(1) 计算每个候选中心点与其他候选中心点的欧式距离，其距离表示为：

$$D = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2} \quad (4-5)$$

其中 (X_1, Y_1) 和 (X_2, Y_2) 是两个候选点的坐标。如上图 4-7 中的(1)所示。

(2) 通过距离阈值为每个候选中心点创建一个集合。对于 (a) 中满足两个候选点的距离小于距离阈值 thd ，将该点划入一个集合中。如上图中的 (b) 所示。

(3) 合并具有公共元素的集合。若两个集合中含有公共元素，则将这两个集合归为一个新的集合；若两个集合中没有公共元素，则两个集合则被认为是毫无相关的两个类。如图中 (c) 所示。

(4) 分别计算各集合的中心点。得到每个钢筋的中心点，统计中心点的个数即可。

获得钢筋是数目即实现计数功能。

本文距离聚类的伪代码实现如下图 4-8 所示：

Algorithm: Distance Clustering (DC)
Input: The location of candidate center points and a distance threshold thd
Output: A set of clusters, the center of each cluster and the number of clusters
Step 1:Initialization
1) For $i = 1, \dots, n$, set $D(i) = d_{ij}$, where n is the number of candidate center points, d_i is the Euclidean distance between the i th candidate center point and its closest candidate center point j , D is the set of d_{ij} .
2) Create an empty structure S to save the initial clusters
3) For $i = 1, \dots, n$, do if $D(i) < \text{thd}$, $S\{i\} = \{i,j\}$; else $S\{i\} = \{i\}$ end for
Step 2:clustering
1) Create a structure $Sc\{1\} = S\{1\}$ to save the final clusters and set the number of clusters as $n_{Sc} = 1$.
2) For $i = 2, \dots, n$,do $n_{Sc} = \text{length}(Sc)$ condition = 0 For $k = 1, \dots, n_{Sc}$,do if $(S\{i\} \cap Sc\{k\}) \neq \emptyset$, do $Sc\{k\} = \{Sc\{k\}; S\{i\}\}$ condition = 1 Break end if end for if condition = 0, do $n_{Sc} = n_{Sc} + 1$ $Sc\{n_{Sc} + 1\} = S\{i\}$ end if end for
Step 3:Obtaining Centers
1) Create an empty center set C to save the centers
3) For $i = 1, \dots, n_{Sc}$,do $x_{ci} = (\max(x_{Sc}\{i\}) + \min(x_{Sc}\{i\})) / 2$ $y_{ci} = (\max(y_{Sc}\{i\}) + \min(y_{Sc}\{i\})) / 2$ $C(i) = (x_{ci}, y_{ci})$ end for

图 4-8 距离聚类的伪代码

对于第 i 个候选中心点，首先计算它与其他候选中心点的欧氏距离，然后选择距离第 j 个候选中心点最近候选点，并将其分配给第 i 个候选中心点。此外，还创建了

一个集合 S 来保存第 i 个候选中心点所有点。距离阈值 thd 用于判断第 j 个候选点是否与第 i 个候选中心点相邻，是否属于 $S\{i\}$ 集合。在论文中，距离阈值 thd 设置为 20。最后创建一个新的集合 Sc 来保存最后的集群。对于 S 中的每个集群 $S\{i\}$ ，如果它与 Sc 中的一个集群 $Sc\{k\}$ 具有相同的元素，则该集群 $S\{i\}$ 将被合并到 $Sc\{k\}$ 中。否则， Sc 将创建一个新的集群来保存 $S\{i\}$ 。对于 Sc 中的每个簇，通过求 x 坐标和 y 坐标的最大值和最小值的平均值来计算每个簇的中心点，实现钢筋端面中心点的定位。

4.4 CNN-DC 算法实验结果及分析

本文中个钢筋端面识别可看成一个二分类问题，数据样本中有正负两个类别，最终的训练完成的 CNN 模型的预测和原始的 Label 的组合有 4 种，分别为：

- (1) TP (True Positive)，真正率，标记的样本为正样本，预测的结果为正样本。
- (2) TN (True Negative)，真假率，标记的样本为负样本，预测的结果为负样本。
- (3) FP (False Positive)，假正率，标记的样本为负样本，预测的结果为正样本。
- (4) FN (False Negative)，假负率，标记的样本为正样本，预测的结果为负样本。

在本文中，选取 ACC（准确率），Precision（精准率），Recall（召回率）和 F1 Score 作为评估卷积神经网络模型的评价指标。

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}, \quad (4-6)$$

表示预测正确的结果占总样本的百分比。

$$Pre = \frac{TP}{TP + FP} \quad (4-7)$$

表示在所有被预测为正的样本中实际为正的样本的百分比。

$$Recall = \frac{TP}{TP + FN} \quad (4-8)$$

又称为查全率，在实际为正的样本中被预测为正样本的概率。

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (4-9)$$

F1 分数同时考虑了查准率 (Precision) 和查全率 (Recall)，让二者同时达到最高。

为了进一步评估本算法在钢筋自动计数和中心定位方面的表现，本文还提出了另外三个适用于本实验指标，分别为偏移量 (Offsets)、计数准确率 (ACCr) 和测试一张图片的时间所花费的时间 (Time)。

$$Offsets = \frac{\sum_{i=1}^N \frac{X_i}{m}}{N} \times 100\% \quad (4-10)$$

其中 $m = 71$ 为钢筋的平均直径，用于进行数据归一化。 X_i 为人工标记的第 i 个中心点与检测到的中心点之间的欧氏距离。用偏移量来评估算法的中心定位性能。

$$Accr = (1 - \frac{N_d - N}{N}) \times 100\% \quad (4-11)$$

其中， $N_d = TP + FP$ 为检测到的中心点个数。 N 是钢筋的实际数量。 $Accr$ 评估的是论文中的 CNN-DC 算法在钢筋计数方面的性能。

为了验证所提出的 CNN-DC 算法在钢筋自动计数和中心定位方面的有效性，论文在此部分进行了两个实验。在第一个实验中，分析了 CNN-DC 在钢筋计数和中心定位方面的性能。在第二个实验中，我们对 DC 算法中的两个参数(卷积核的步长和像素距离阈值 thd) 进行了分析。

(1) 第一个实验：CNN-DC 算法在钢筋的性能表现如下表 1 所示，从它可以观察到 CNN-DC 在召回率，精准度和 F1 分数可以获得较高的数值，平均值分别为 0.9951, 0.9976 和 0.9963。这表明 CNN-DC 算法可以有效地识别钢筋图像中心点。此外，CNN-DC 算法可以获得较高的 Accr 分数，其平均值为 99.26%，这说明本文中提出的 CNN-DC 算法在钢筋计数方面可以有很好的表现。此外，较低的偏移值 $offsets$ 为 4.1%表明 CNN-DC 算法在中心定位方面表现良好。CNN-DC 的计算时间 $Time=3.5862$ 表明，CNN-DC 算法能够满足工厂自动化实时处理的要求。

表 4-1 CNN-DC 算法在识别钢筋上的性能表现

CNN-DC 评价指标	Recall	Precision	F1	Accr	Offsets	Time(s)
数值	0.9951	0.9976	0.9963	0.9926	4.1%	3.5862

(2) 第二个实验：在我们的第二个实验中，我们使用了滑动窗口的步长 $\text{stride} = 6$ 和像素之间的距离阈值 $\text{thd} = 20$ ，这两个参数是基于我们对钢筋数据集的训练数据进行实验后选择的。为了说明选择这些值的原因，图 4-9 和图 4-10 分别给出了通过改变步长 stride 和像素距离阈值 thd 对钢筋数据集的训练数据的查全率、精密度、F1、偏移量和计算时间等评价指标的影响。

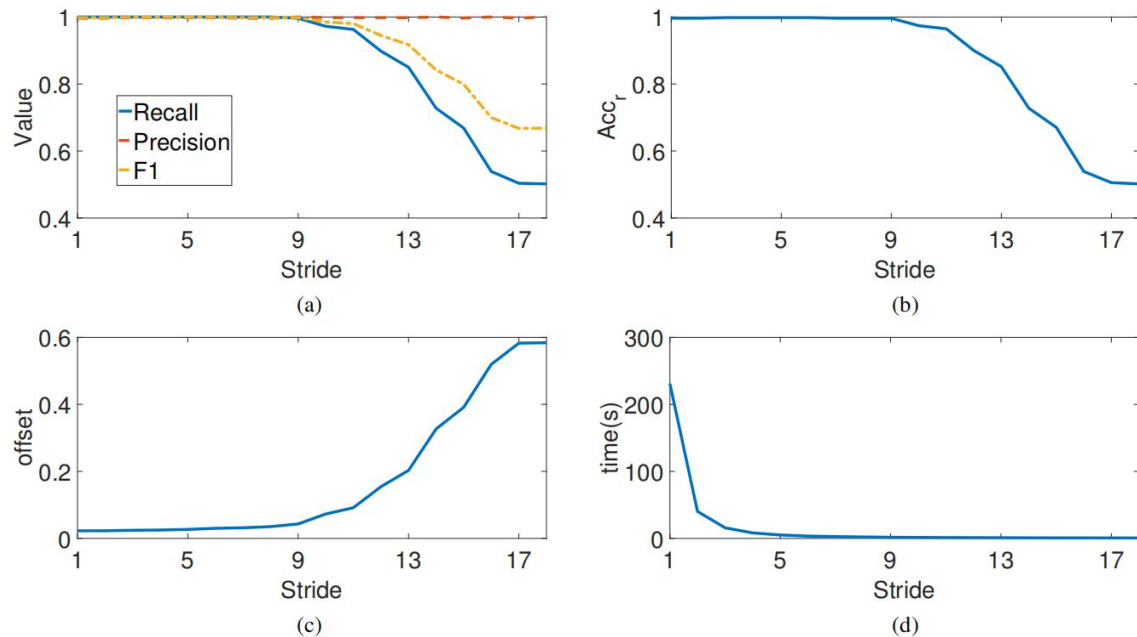


图 4-9 不同的评价指标受随滑动窗口的步长的影响

图 4-9 中的(a)图是查全率、查准率、F1 随步幅增加的变化。(b)图是 Accr 随步幅增加的变化。(c)偏移量随步幅增加的变化。(d)算法测试的运行时间随步幅增加的变化。

从图 4-10 可以看出，当步长处于[1,9]区间的时候，CNN-DC 算法对钢筋图片训练出的网络模型的识别效果最好，当步长处于[5,17]区间的时候，网络模型测试图片所花费的时间最短。因此滑动窗口的步长应该在[5,9]区间进行选取，在本论文中选取的步长 stride 为 6。

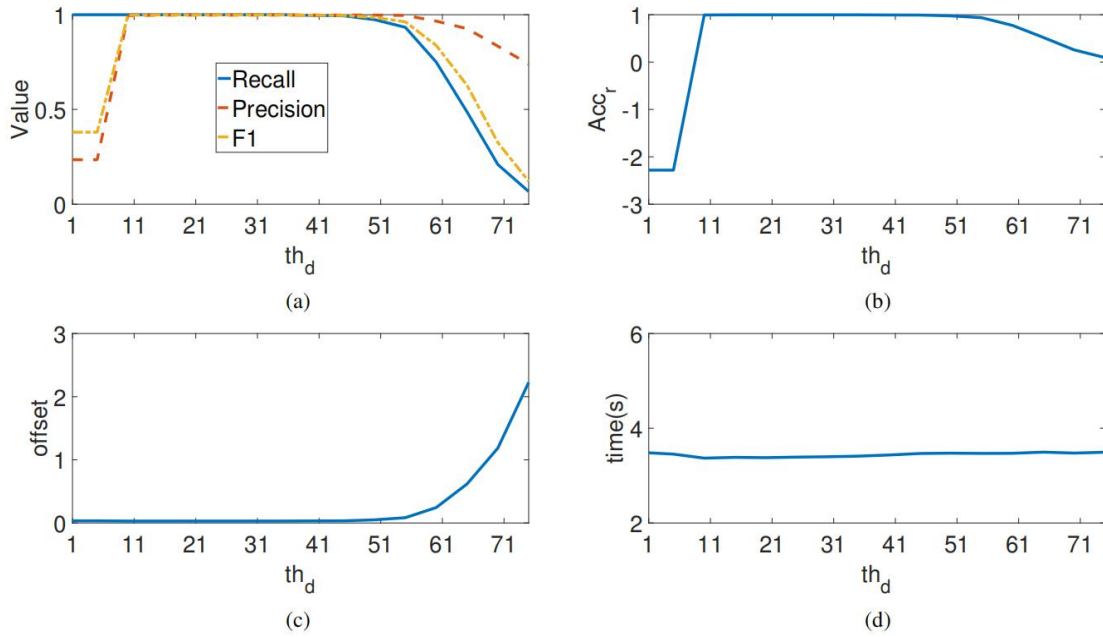

 图 4-10 不同的评价指标受像素距离阈值 th_d 的影响

图 4-10 中的 (a) 表示查全率、查准率和 F1 随距离阈值 th_d 的增加而变化。(b) 表示随着距离阈值 th_d 的增加, $Accr$ 的变化情况。(c) 表示偏移量 $offsets$ 随距离阈值 th_d 增加的变化。(d) 表示模型测试时间随 th_d 的增加而变化。

从图 4-10 可以看出, 随着距离阈值 th_d 值处于 $[11, 51]$ 区间时, CNN-DC 算法可以保持较高的查全率、查准率、F1 和 $Accr$ 值。当距离阈值 th_d 在 $[1, 51]$ 中变化时, CNN-DC 算法可以一直保持一个较低的偏移值。而模型测试图片所花费的时间受距离阈值 th_d 变化影响很小。因此由以上可知, 当距离阈值 th_d 处于 $[11, 51]$ 时, CNN-DC 算法可以获得较好的性能。在本论文中, 选取距离阈值 th_d 为 20。

钢筋端面的中心点检测和计数在钢筋工厂的自动化生产中具有重要意义。钢筋计数和中心定位传统上由熟练工人来完成, 既繁琐又耗时。为了减轻工人对钢筋计数和中心定位的负担, 本文提出了一种有效的钢筋计数和中心定位的 CNN-DC 算法。论文中提出的 CNN-DC 算法首先利用深度卷积神经网络进行钢筋端面的候选中心点检测, 然后利用距离聚类算法对候选中心点进行聚类, 得到钢筋的中心位置。实验结果表明, 提出的 CNN-DC 框架在钢筋计数和中心定位方面表现良好, 查全率 $Recall = 0.9951$, 精准率 $Precision = 0.9976$, $F1 = 0.9963$, 计数准确率 $Accr = 99.26\%$, 偏移量 $Offsets = 4.11\%$, 时间 $Time = 3.5862 (s)$ 。

第 5 章 基于 CNN 的密度估计实现钢筋端面中心检测及计数

本文第 4 章中提出的 CNN-DC 算法中, 滑动窗口中窗口的大小、滑动的步长的选取对钢筋端面中心点检测以及计数的准确率影响较大, 因此 CNN-DC 算法只能应用于固定角度、距离拍摄的钢筋图片, 且滑动窗的操作耗时较大, 实时性较差。为了解决这一问题, 第二种基于 CNN 的密度估计实现钢筋中心点检测和计数的方法可以处理不同角度, 不同距离的钢筋图片且能实现端到端的训练。

人群数目统计是一项具有挑战性的任务, 现实场景中由于人群中严重的遮挡、场景视角扭曲和人群分布的多样性, 导致行人检测和跟踪的实现在人群场景中应用比较困难, 目前最先进的方法大多是基于回归的, 目的是学习底层特征与人群计数之间的映射关系。然而, 这些算法是用来解决特定场景的人群数目统计。针对特定场景学习的人群计数模型只能应用于同一场景, 当给定一个看不见的场景或更改的场景布局, 必须使用新的标记数据对模型进行重新训练。

人群计数是一种计算机视觉任务, 它将人群图像作为输入, 输出相应的人群密度图, 最后对该图进行求和, 得到最终的行人数量。近年来, 随着深度学习的兴起, 人群计数得到了迅猛的发展。一方面, 近年来出现了许多带有人工标注的大型数据集, 另一方面, 许多基于 CNN 的模型被开发出来, 如 MCNN、CSRNet、SANet^[45]等。

本文受人群密度估计算法的思想的启发, 利用一个开源的人群计数代码框架 (C-3-Framework)^[46], 该框架是在 Pytorch 上开发的, 它是一个高效的、可靠的人群计数任务开发工具包。另外搭建了一个 CNN 卷积神经网络。网络的输入为钢筋图片, 输出为 CNN 卷积神经网络预测的密度图 (Density Map), 通过对密度图的积分得到图片中钢筋的数目。该网络用于处理样本数据充足的情景, 且能实现端到端的训练。下图所示为本文基于卷积神经网络的密度估计实现钢筋端面中心点检测及计数算法的流程框架。

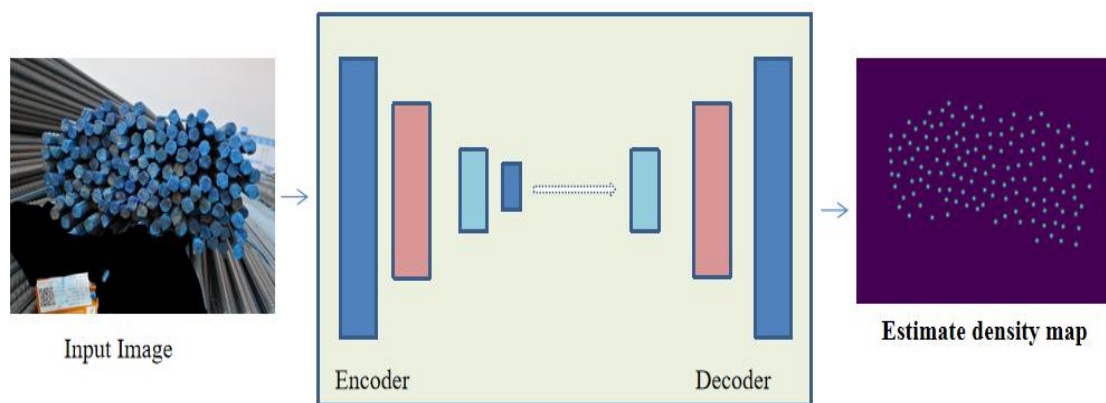


图 5-1 基于卷积神经网络的密度估计算法处理流程

5.1 数据预处理

本文中的数据图片来自于 2019 数字中国创新大赛即《智能盘点—钢筋数量 AI 识别》比赛，是由广联达公司提供的钢筋进场现场的图片。如下图 5-2 所示，图示展示的是本文训练 CNN 卷积神经网络所用的数据集。该数据集中共有 250 张图片，图片中的钢筋均为捆包装，图片中钢筋的直径变化范围较大且截面形状不规则、颜色不一，拍摄的角度、距离也各不相同。

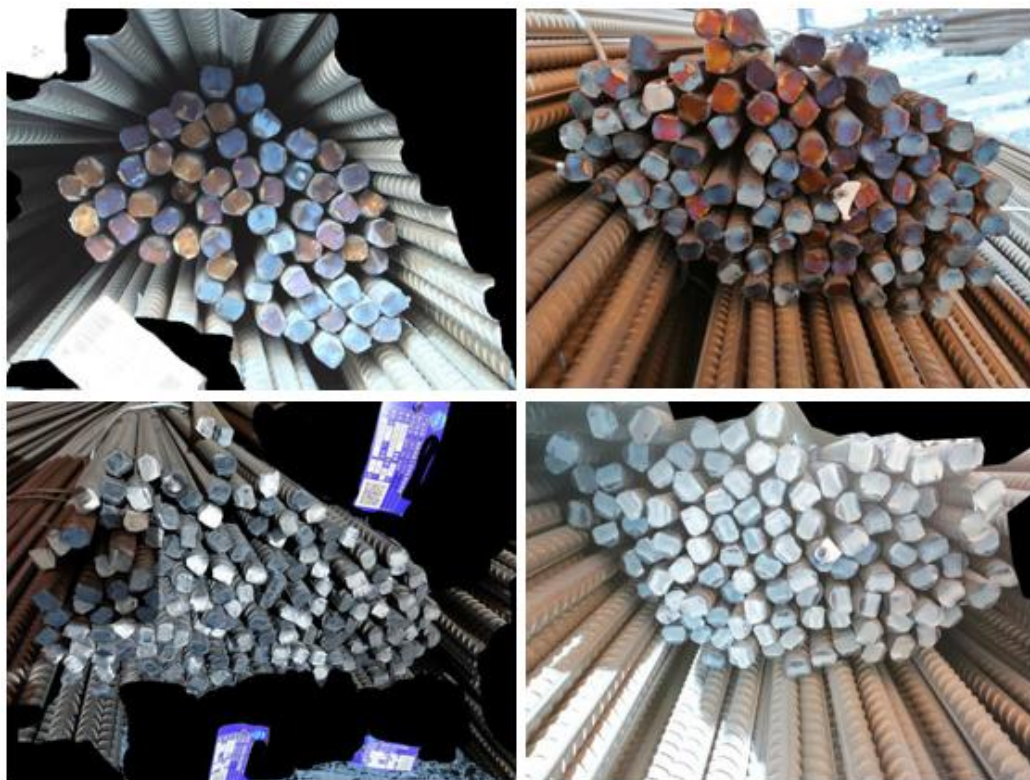


图 5-2 展示的是本文训练 CNN 卷积神经网络所用的数据集

在官方所提供的数据集基础之上，本文中使用了图像增强的方法来扩充我们训练卷积神经网络所需要的数据集。目的则是增加我们样本空间的多样性，增加训练出来后模型的泛化性能。

图像增强是扩充数据样本规模的一种常见的方法。深度学习是一种基于大数据训练的方法，数据的规模越大，图像的质量越好，最终的模型则具有更好的识别和分类性能。实际采集的钢筋图片数据会受到光线、拍摄角度等问题，即所拍摄的图片数据很难包括了所有的应用场景。因此在训练网络模型的时候，需要加入光照变化等方面的图像增强，使得训练的模型更能适应不同的应用场景。图像增强可以分为空间几何变换（如水平或垂直翻转，随机裁剪、旋转等）、像素颜色变换（如往图片数据中添加高斯噪声、椒盐噪声等）、高斯模糊、随机擦除以及图像锐化等。本文中采取了空间几何变换、像素颜色变换等对原始数据集进行了图像增强。

本文中对原始数据集采用了图像空间的几何变换即图像随机旋转、错切边换、透视变换等。其中在对原始图片随机旋转中，本文中设置的随机左右旋转角度的数值为 20。在原始图片中进行透视形变，本文设置的描述上下左右方向的垂直型变的程度参数设置为 0.2。对原始图片进行平移变换，其中平移的尺度设置为 10。经过数据扩充后，训练卷积神经网络的数据集共有 1050 张图片。下图 5-3 所示为图像增广后的部分结果。

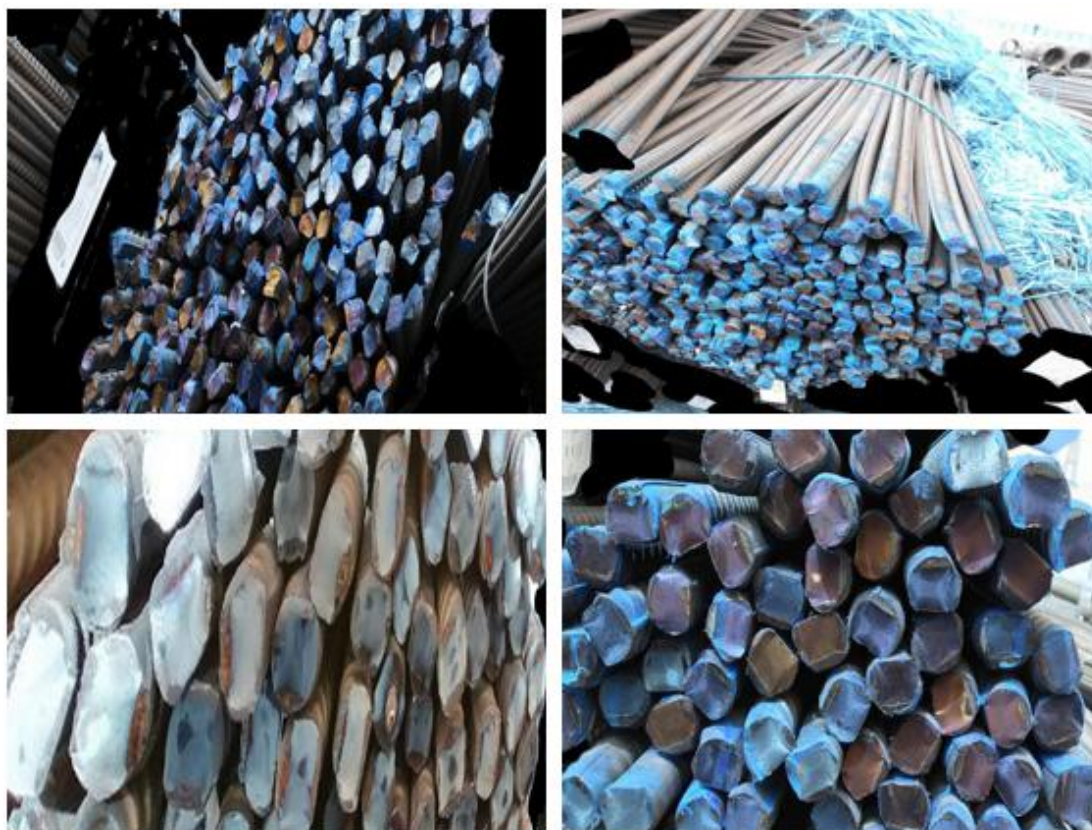


图 5-3 所示为数据增广后的部分图片

在经过数据增广后的数据集中，人工对图片中钢筋的中心点进行标记，标记的样本点为红色点即标记的像素点对应的 RGB 通道值为 (255, 0, 0)。在此基础之上在每个钢筋端面人工标记点使用高斯滤波器对其进行处理，即利用二维高斯分布函数，生成高斯模板，然后用该模板取扫描图像块的每一个像素值（本文中图像块所选取的是距离标记点 5 个像素点距离的圆形区域），用高斯模板确定的领域内像素的加权平均值作为新图像中模板中心位置的像素值。本文中采用的二维高斯分布函数为：

$$G(X,Y)=\frac{1}{2\pi\sigma^2}e^{-\frac{X^2+Y^2}{2\sigma^2}} \quad (5-1)$$

其中， σ 为高斯函数里面的标量或标量序列，这个值越大，表示滤波之后的图像越模糊。本文中的 $\sigma=5$ 。计算平均值的时候，我们只需要将中心点作为原点，其他点按照其在标准正态曲线上的位置上分配权重，就可以得到一个加权平均值。下图 5-4 所示为我们的标记数据以及利用二维高斯函数在标记点周围生成的最终样本图片。

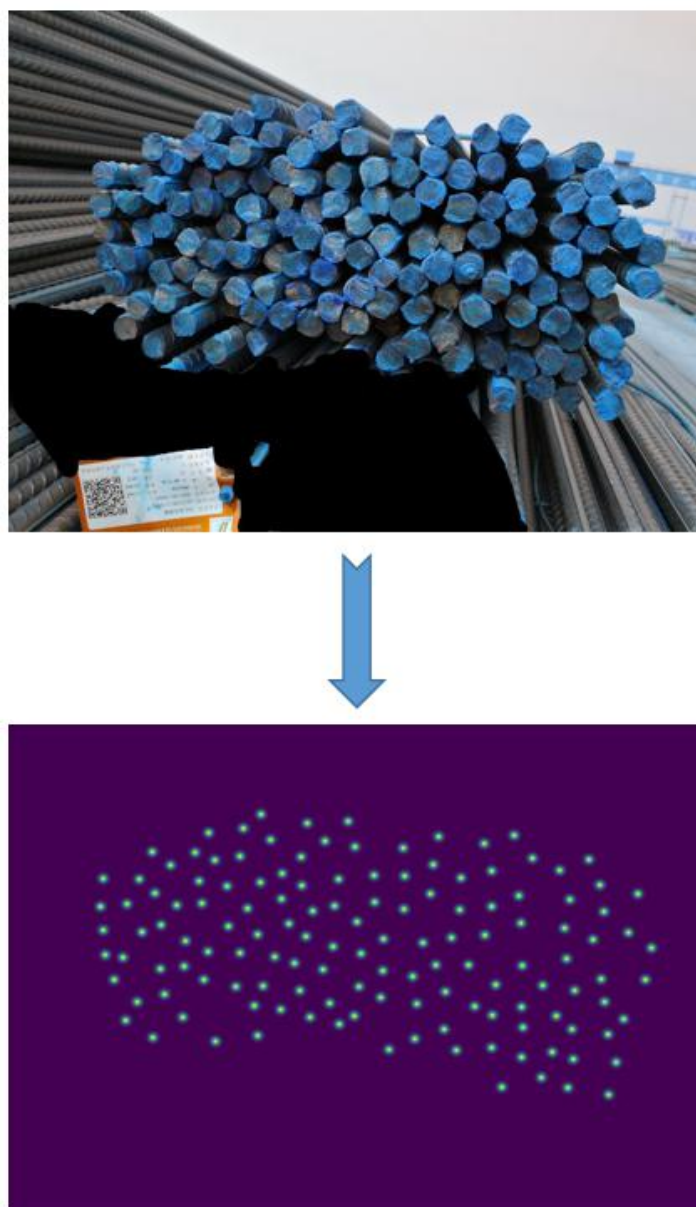


图 5-4 原始样本数据经过人工标记后，形成的高斯密度图

本文处理完成的图片数据按照训练集：测试集：验证集 = 6:2:2 的比例划分，因此训练集、测试集和验证集的图片数量分别为 630，210，210 张。模型的训练过程中使用交叉验证法，尝试利用不同的训练集/验证集划分来对模型做多组不同的训练/验证，该方法用来应对单独测试结果过于片面的问题。

5.2 基于 CNN 的密度估计的网络搭建

本文的基于 CNN 的密度估计搭建过程可分为两个阶段，分别为图像数据预处理阶段和钢筋中心点检测和计数模型。数据预处理策略主要由两部分组成，即输入图片

尺寸大小和对应 Ground Truth 的变换。其中对于输入图像大小，本文中我们限制输入图像的高度和宽度，以确保它们能被 16 整除。这个限制保证了一些向下采样层(比如 max-pooling)可以输出我们想要的大小。最终网络的输出密度图的尺寸原始图像的 1/8。本文使用上采样层来匹配输出和输入图像的大小相等。

本文的卷积神经网络由编码器和解码器组成，其中编码器由 4 个卷积层组成，保证图像的特征能够被充分提取出来。解码器由两个卷积层和一个上采样层组成。本文使用的 CNN 的详细信息见下图 5-5 所示。

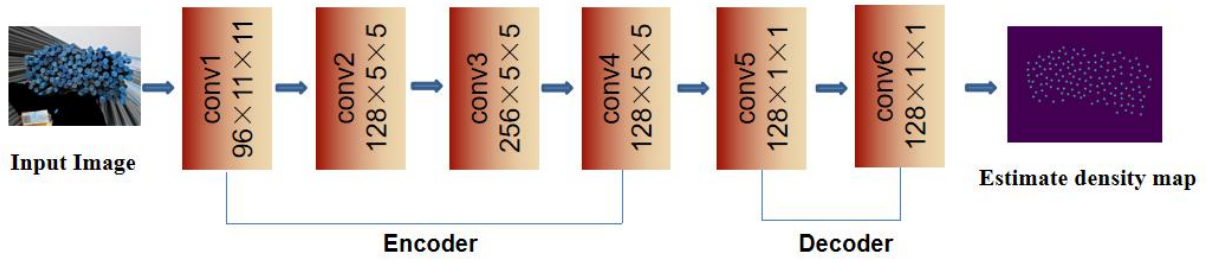


图 5-5 为本文使用卷积神经网络的结构

卷积神经网络的输入为灰度图片，下图 5-6 详细的描述了搭建 CNN 的结构：

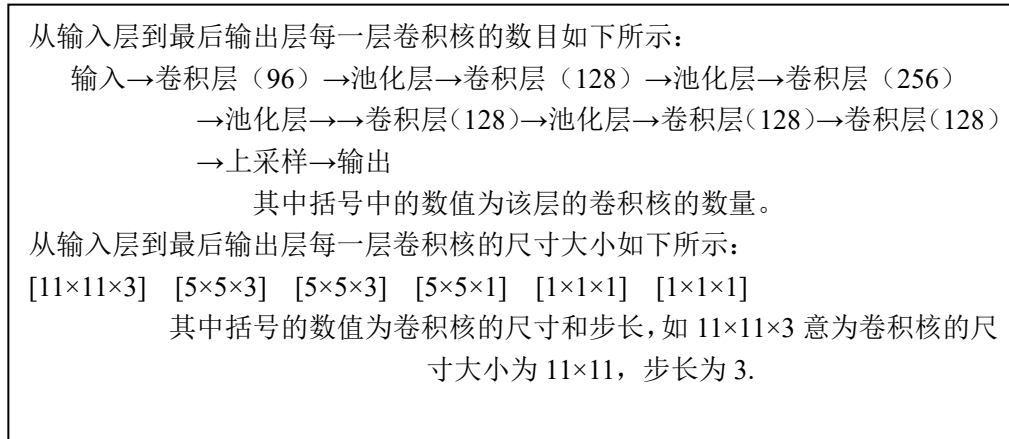


图 5-6 搭建的 CNN 的每一层的参数

我们修改了其在 conv1 和 conv2 中的填充操作，以保证生成的 Feature map 可以被正常分割，并且使用 4 层的卷积层作为图像特征编码器，其中输出比例为原始图像比例的 1/16。解码器由两个卷积层和一个上采样层组成，直接回归最终的 1 通道密度图。通过对 CNN 的训练，以最小化模型预测的密度估计图和 Ground Truth 的密度图之间的欧式距离。损失函数如下所示：

$$L(\theta) = \frac{1}{2N} \sum_{i=1}^N \|Dx_i(\cdot; \theta) - D_{xi}^{GT}(\cdot)\|_2^2 \quad (5-2)$$

$Dx_i(\cdot; \theta)$ 表示第 i 张输入图片对应的 CNN 回归输出的密度图。 $D_{xi}^{GT}(\cdot)$ 表示第 i

张输入图片对应的 Ground Truth 密度图。N 是训练样本的数量。本文中模型的初始学习率为 0.001，学习率衰减率为 0.095，权重衰减对于模型的学习是重要的，权重衰减不仅仅是一个正则项：它减少了模型的训练误差。训练的最大的迭代层数为 1000 个 Epoch。

卷积神经网络的训练通过随机梯度下降反向传播来优化损失函数 $L(\theta)$ ，在求解损失函数的最小值时，可以通过梯度下降法来一步步的迭代求解，得到最小化的损失函数和模型参数值。在这里， $L(\theta)$ 损失函数作为回归估计计数和真实计数之间的计数误差，通过它间接地最小化计数误差。

5.3 模型的评估

本文中使用平均绝对值误差(MAE)和平均均方误差(MSE)作为衡量指标来比较卷积神经网络的计数性能。对于 N 幅图像的测试序列，平均均方误差 MAE 定义如下：

$$MAE = \frac{1}{N} \sum_{i=1}^N |C_i - C_i^{GT}| \quad (5-3)$$

平均绝对值误差 MSE 定义如下：

$$MSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (C_i - C_i^{GT})^2} \quad (5-4)$$

其中 C_i 为待评价模型预测的钢筋数量， C_i^{GT} 为人工标注的钢筋数量。MAE 是预测整个测试图片的钢筋计数准确性的指标。MSE 是 MAE 的一个补充度量，用来表示卷积神经网络模型预测计数功能的鲁棒性。下图为卷积神经网络测试的结果示意图。

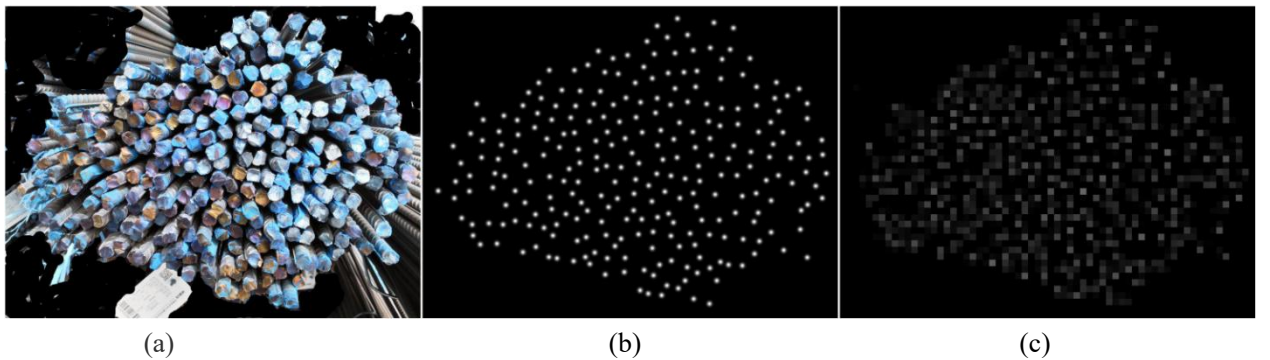


图 5-7 测试单张图片结果示意图

其中图 5-7 中的(a)为测试图片,(b)为所训练模型预测测试图片中钢筋中心点的位置图片, (c)为所训练模型预测测试图片中的概率密度图片

下图 5-8 为模型的训练过程中, 平均绝对值误差 (MAE) 和平均均方误差 (MSE) 随训练 Epoch 的变化过程。一个 Epoch 是当一个完整的数据集通过了神经网络一次并且返回了一次, 即所有训练样本在神经网络中都进行了一次正向传播和一次反向传播。一个 Epoch 就是将所有训练样本训练一次的过程。

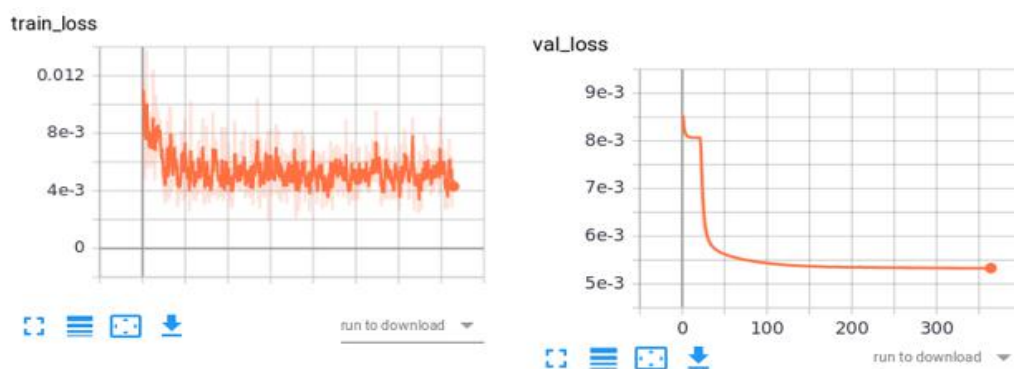


图 5-8 平均绝对值误差(MAE)和平均均方误差(MSE)随训练 Epoch 的变化过程

下图为模型的训练过程中, 训练误差(train_loss)和评估误差(val_loss)随训练 Epoch 的变化过程。

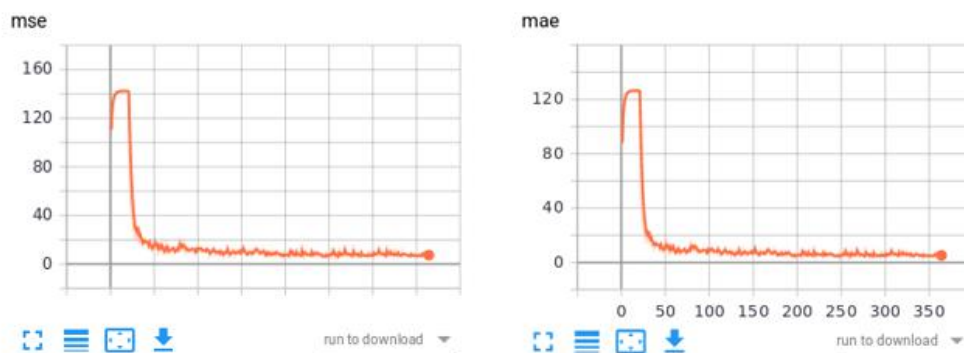


图 5-9 训练误差(train_loss)和评估误差(val_loss)随训练 Epoch 的变化过程

由上图可以看出, 当 Epoch=50 的时候, 平均绝对值误差(MAE)和平均均方误差 (MSE)曲线 1 趋于稳定。当 epoch=200 的时候, CNN 模型的训练误差和评估误差趋于稳定, 此时 CNN 模型处于收敛状态。

除此之外, 本文中也使用了一些常用分类网络如 AlexNet、VGG 和 ResNet 系列与我们的方法进行对比。类似的在 AlexNet 中, 使用了前 5 层卷积神经网络提取图像的特征, 后面使用了两层卷积层和上采样输出预测密度图。本文中对 VGG 进行了修改, 采用了 VGG 的前 10 个卷积层作为编码器, 后面使用了三个反卷积层输出预测密度图。同样的, 本文在 ResNet 中为了保持最终密度图的尺度, 我们将第三层的

的步长由 2 改为 1，解码器部分器由两个反卷积层组成和上采样组成。下表 2 为使用 MAE 和 MSE 两个评价指标在不同的网络中表现。

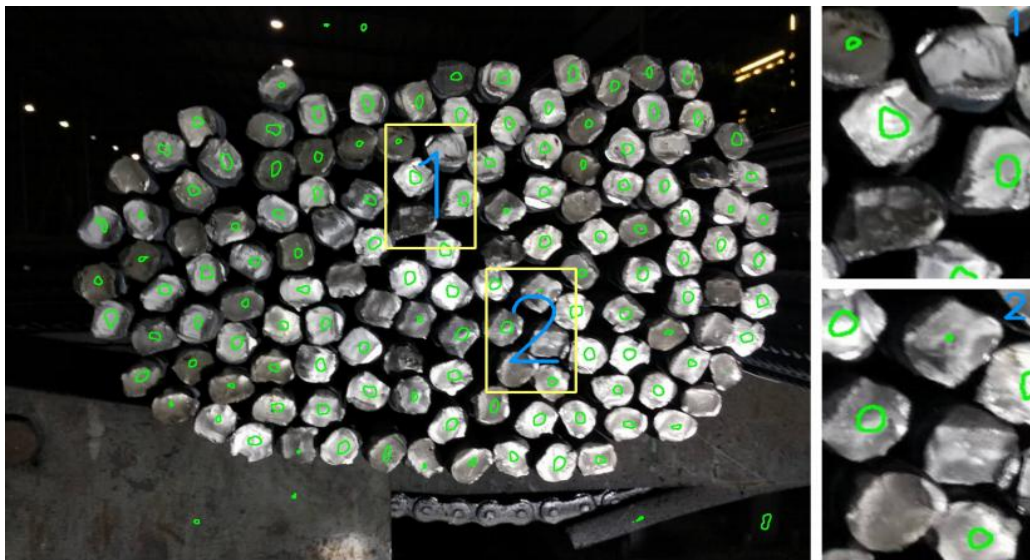
表 5-1 MAE 和 MSE 两个评价指标在不同的网络中表现

网络模型	MAE	MSE
AlexNet	5.72	8.91
VGG	7.3	9.5
ResNet	7.7	10.6
Proposed	4.61	6.84

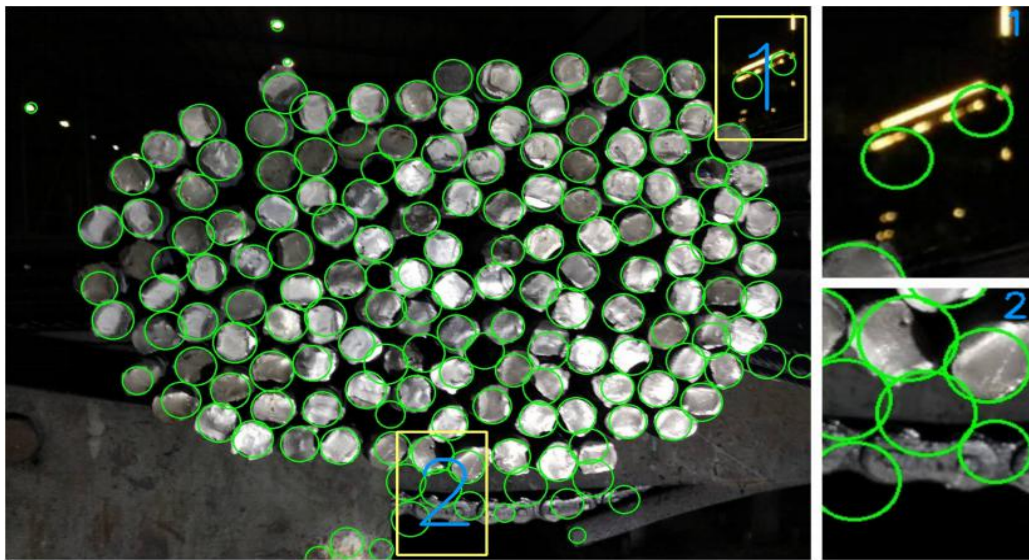
由上表 5-1 可以看出，本文搭建的 CNN 网络的平均绝对值误差(MAE)和平均均方误差(MSE)分别为 4.61 和 6.84。相比于 AlexNet、VGG 以及 ResNet 网络，本文基于 C-3-F 框架搭建的 CNN 网络测试的指标表现力更好。

5.4 实验结果对比和总结

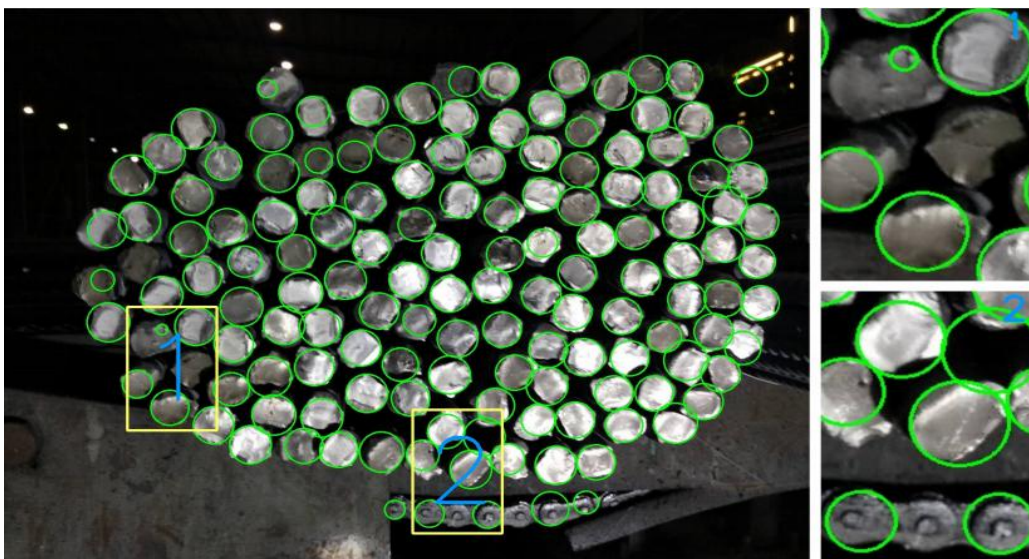
为了进一步证明本文提出的 CNN-DC 算法和基于 CNN 的密度估计算法的有效性，我们将这两种算法与钢筋数据集上的其他现有方法进行了比较。下图 5-9 为传统图像处理方法与本文提出的两种算法测试同一张图片的中间实验结果对比图。



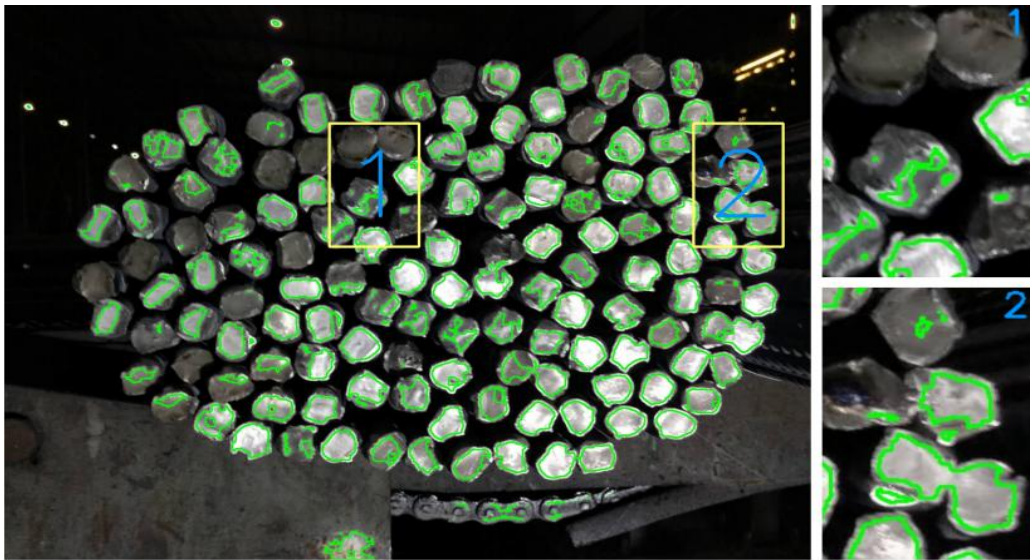
(a) Zhang et.al



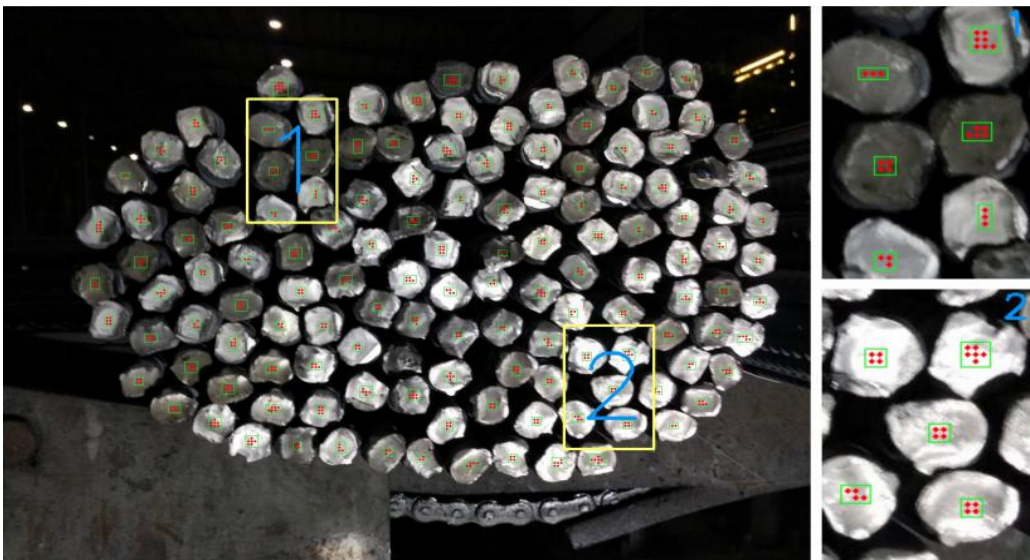
(b) Ying et.al



(c) Ghazali et.al



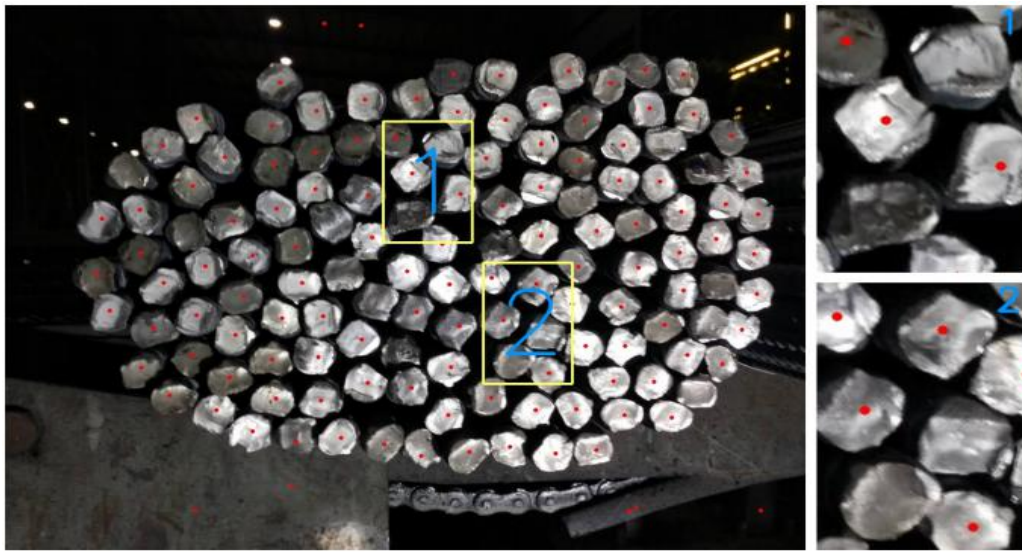
(d) Liu et.al



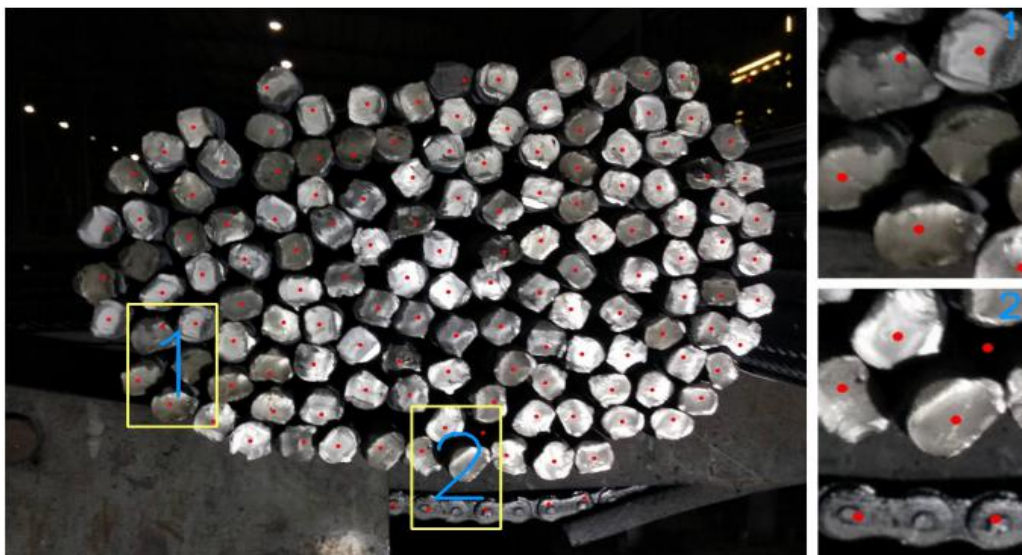
(e) Proposed1

图 5-9 传统图像处理方法与本文提出第一种算法测试同一张图片中间实验结果对比图

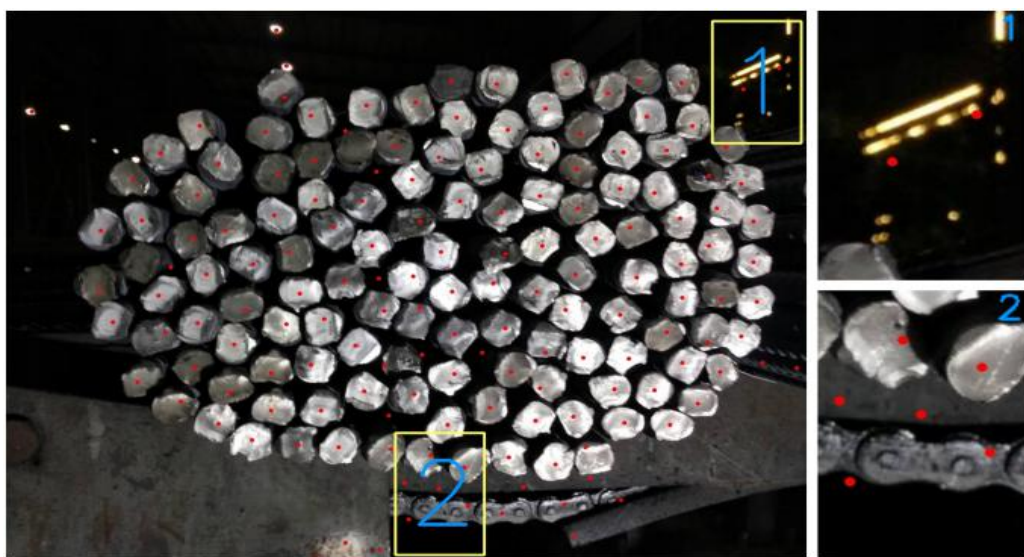
下图 5-10 为传统图像处理方法与本文提出的两种算法测试同一张图片的最终实验结果对比图。



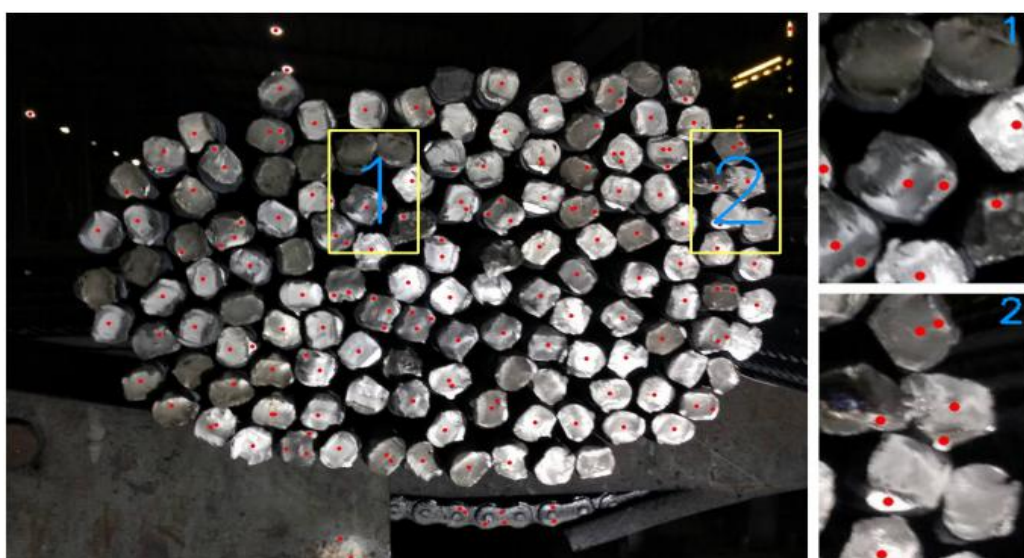
(a) Zhang et.al



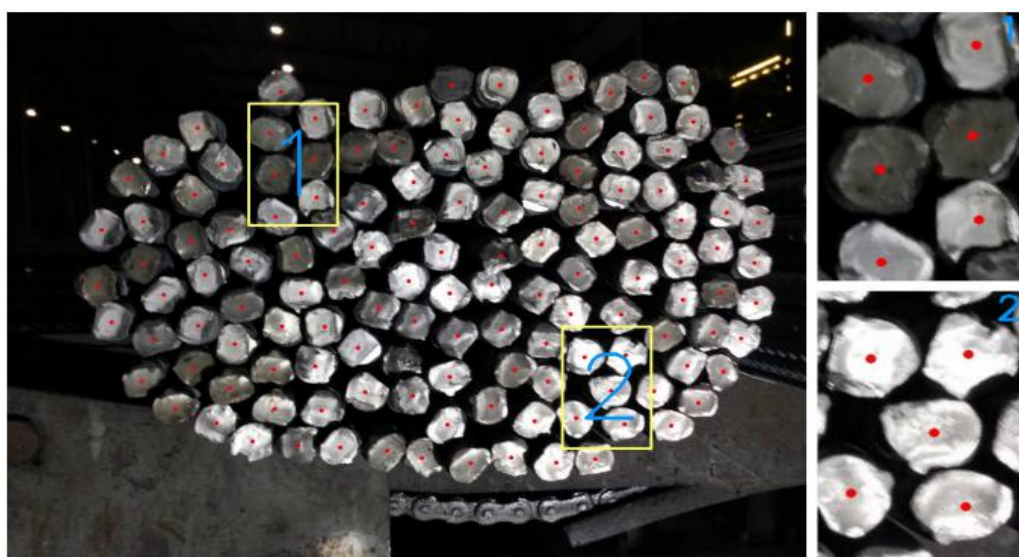
(b) Ying et.al



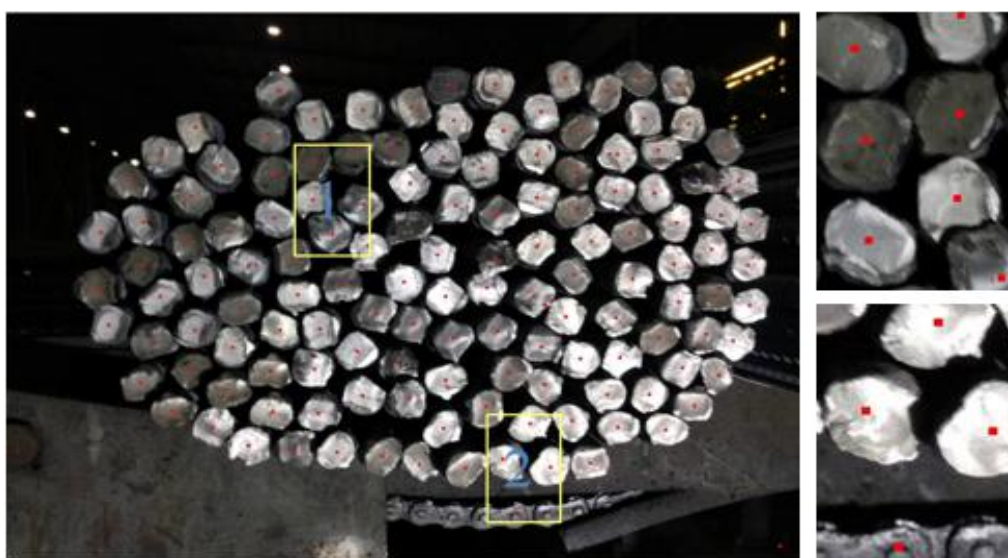
(c) Ghazali et.al



(d) Liu et.al



(e) Proposed1



(f) Proposed2

图 5-10 传统图像处理方法与本文提出的两种算法测试同一张图片最终实验结果对比图

在上面的测试结果图片中可以看出。在 Zhang et.al 所提出的算法^[7]中，由于钢筋端面的形状各不相同，模板匹配算法无法匹配与模板不相似的图像，并且识别粘连的钢筋端面。Ying et.al 等人提出的算法中，霍夫变换对钢筋的边缘信息较为敏感因此将背景区域的部分图片误识别为钢筋。Ghazali 等人^[10]提出的识别圆形和矩形钢筋识别的算法中，本测试图片的一些圆形的背景区域被误识别为钢筋端面。在 Liu et.al 等人中提出的基于轮廓的算法中，该算法依赖于前期对钢筋轮廓良好的提取，对钢筋表面的亮度变化和钢筋的边缘模糊敏感，导致一些单钢筋错误地标识了多根钢筋，而有些多根钢筋错误地标识了单根钢筋，因此出现识别的中心点位置处于两根钢筋接触

的边缘，无法有效的识别钢筋端面的中心点。

如下表 5-2 所示，为进一步比较算法之间的表现，我们采用了以下六种评估指标。分别是召回率(Recall)、精准率(Precision)、F1 分数(F1 Score)、计数准确率(Accr)、偏移量(Offset)、运行时间(Time)。

表 5-2 六种评估指标在不同算法上的对比

Average	Recall	Precision	F1	Accr	Offsets	Time(s)
Zhang.et.al	0.8864	0.9360	0.9103	94.69%	15.83%	0.3023
Ying.et.al	0.9617	0.8417	0.8975	85.68%	12.58%	0.2404
Ghazali.et.al	0.9778	0.9366	0.9566	95.56%	10.30%	0.1346
Liu.et.al.	0.8123	0.6833	0.7420	80.99%	25.58%	0.0313
Proposed1	0.9951	0.9976	0.9963	99.26%	4.11%	3.5862
Proposed2	0.9626	0.9435	0.9530	94.03%	5.62%	0.4026

由上述表格数据结果可以看出，本文提出的两种算法在召回率、精准率、F1 分数、计数准确率、偏移量等指标的变现均比传统图像处理方法更好，其中 CNN-DC 算法（Proposed1）在召回率、精准率、F1 分数以及计数准确率平均值在 0.99 以上。而基于 CNN 的密度估计算法中（Proposed2），上述指标均优于传统图像处理方法，略低于 CNN-DC 算法，但是在实时性方面优于 CNN-DC 算法。同时经分析可以看出本文提出的两种算法在对环境的干扰，钢筋端面亮度的变化和钢筋边缘模糊方面具有良好识别效果。

第 6 章 总结及下一步工作

6.1 论文总结

钢筋的中心点检测以及计数的准确率对钢筋生产有着重要的意义,利用数字图像处理技术实现钢筋端面中心点检测和计数,配合机械设备实时的帮助一线工人降低他们的劳动强度和安全隐患,已经成为一个迫切解决的问题。本文的主要贡献如下:

1、针对应用场景提出了两种不同的解决方法,第一种算法首先用 CNN 检测钢筋端面候选中心点。在此基础上,提出了一种有效的聚类算法——距离聚类(DC)算法来聚类候选中心点定位钢筋的中心。

2、第二种方法通过编码器和解码器得到单一特征图,对预测的密度图进行积分实现钢筋端面中心点的检测和计数功能。该算法能实现网络端到端的训练,且对不同场景下的钢筋图片具有很好的识别准确率

3、通过将本文提出的两种方法与传统方法进行对比,可以发现本文中的两种方法在识别钢筋端面中心的偏移量,计数准确率等指标上均比传统方法的表现力更好。

4、本文中两种算法的联系: CNN-DC 算法对于固定角度、距离拍摄的钢筋图片的识别精度较高,但是实时性较差。为了解决这一问题,第二种基于 CNN 的密度估计实现钢筋中心点检测和计数的方法可以处理不同角度,不同距离的钢筋图片且能实现端到端的训练。虽然识别的精度不如 CNN-DC 算法但是在实际工厂环境中可以牺牲一些精度来达到实时性的要求。

6.2 下一步工作

1、在 CNN-DC 算法中, Patch 的尺寸大小关系到其识别的性能,因此下一步工作中,我们可以先使用本文中的第二种算法基于 CNN 的密度估计得到 Patch 尺寸大小的估值,在此基础之上使用 CNN-DC 算法,这样既能满足识别不同角度、距离的钢筋图片,又进一步提高了算法的性能。

2、在 CNN-DC 算法中,运行时间比其他方法要长,但可以通过后续对网络模型裁剪/剪枝^[43,44]、网络二值化^[45]等方法来减少计算时间。

参考文献

- [1] Kajii Sadao , Umehara Satoru , Ouchi Tomoji. Automatic Separating Device for Bar Steel Continuously Transported by Chain Conveyer [P]. 日本专利: JP57126315A, 1982-08-06.
- [2] M. F. Ghazali, LK. Wong, and J. See, “Automatic detection and counting of circular and rectangular steel bars,” in 9th International Conference on Robotic, Vision, Signal Processing and Power Applications. Springer, 2017, 199–207.
- [3] D. Zhang, Z. Xie, and C. Wang, “Bar section image enhancement and positioning method in on-line steel bar counting and automatic separating system,” in Image and Signal Processing, 2008. CISP’08. Congress on, vol. 2. IEEE, 2008, 319–323.
- [4] X. Ying, X. Wei, Y. Pei-xin, H. Qing-da, and C. Chang-hai, “Research on an automatic counting method for steel bars’ image,” in Electrical and Control Engineering (ICECE), 2010 International Conference on. IEEE, 2010, 1644–1647.
- [5] L. Xiaohu and O. Jineng, “Research on steel bar detection and counting method based on contours,” in 2018 International Conference on Electronics Technology (ICET). IEEE, 2018, 294–297.
- [6] Yang Wu, Xiaofeng Zhou, Yichi Zhang. Steel bars counting and splitting method based on machine vision[C]. IEEE International Conference on Cyber Technology in Automation. IEEE, 2015.
- [7] Cong Zhang, Hongsheng Li, Xiaogang Wang. Cross-scene crowd counting via deep convolutional neural networks[C]. 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2015.
- [8] Zhang Y , Zhou D , Chen S , et al. Single-Image Crowd Counting via Multi-Column Convolutional Neural Network[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2016.
- [9] Yuhong Li, Xiaofan Zhang, Deming Chen. CSRNet: Dilated Convolutional Neural Networks for Understanding the Highly Congested Scenes[C]. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Ranjan, Viresh, Le, Hieu, Hoai, Minh. Iterative Crowd Counting[J]. 2018.
- [10] Gao, Junyu, Lin, Wei, Zhao, Bin, C^3 Framework: An Open-source PyTorch Code for Crowd Counting[J]. 2019.

- [11] Alex Krizhevsky, I Sutskever, G Hinton. ImageNet Classification with Deep Convolutional Neural Networks[C]. NIPS. Curran Associates Inc. 2012.
- [12] Simonyan, Karen, Zisserman, Andrew. Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer ence, 2014.
- [13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Deep Residual Learning for Image Recognition[C]. IEEE Conference on Computer Vision & Pattern Recognition. IEEE Computer Society, 2016.
- [14] 姜贺. 基于几何特征的人脸识别算法的研究[D]. 大连理工大学.
- [15] 李文琴, 曾广朴, 文俊浩. 基于神经网络的人脸识别系统[J]. 激光杂志, 2015, 36(3):54-57.
- [16] 姜炳旭, 刘杰, 孙可. Sobel 边缘检测的细化[J]. 沈阳师范大学学报:自然科学版, 2010, 028(004):503-506.
- [17] 王卜堂, 杨善林. 基于 Gauss-Laplace 算子的灰度图像边缘检测[J]. 计算机工程与应用, 2003(26):132-134.
- [18] John F. Canny. Some algebraic and geometric computations in PSPACE[C]. Proceedings of the 20th Annual ACM Symposium on Theory of Computing, 1988.
- [19] Nobuyuki Otsu. A Threshold Selection Method from Gray-Level Histograms[J]. IEEE Transactions on Systems Man & Cybernetics, 2007, 9(1):62-66.
- [20] 王茜蓓, 彭中, 刘莉. 一种基于自适应阈值的图像分割算法[J]. 北京理工大学学报, 2003, 23(4):521-524.
- [21] 梅林, 吴立德, 王裕文. 脉冲加热红外无损检测中的图像处理[J]. 红外与毫米波学报, 2002(05):54-58.
- [22] 胡利平. 合成孔径雷达图像目标识别技术研究[D]. 西安电子科技大学, 2005.
- [23] 秦红星, 蔡绍哲. 彩色血液细胞图像的分割[J]. 中国血液流变学杂志, 2003, 13(3):281-285.
- [24] 聂生东, 章鲁, 顾顺德等. 磁共振图像的分割[J]. 国外医学:生物医学工程分册, 1999(6):349-355.
- [25] 应义斌. 水果图像的背景分割和边缘检测技术研究[J]. 浙江大学学报:农业与生命科学版, 2000(01):37-40.
- [26] 单琳子. 基于模板匹配技术的物品图像高效快速识别算法研究[D]. 北京邮电大学, 2015.

- [27] Xiaoyu Wang, Tony X. Han, Shuicheng Yan. An HOG-LBP human detector with partial occlusion handling[C]. IEEE International Conference on Computer Vision. IEEE, 2009.
- [28] G. Storch. The Neogene mammalian faunas of Ertemte and Harr Obo in Inner Mongolia (Nei Mongol), China. 11. Soricidae (Insectivora)[J]. Senckenbergiana lethaea, 1995, 75(1):221-251.
- [29] RATSCH,G. Soft margins for adaboost[J]. Machine Learning, 2001, 42(3):287-320.
- [30] Joachims, Thorsten. Making large-scale SVM learning practical[J]. Technical Reports, 1998, 8(3):499-526.
- [31] Lecun, Y, Bottou, L. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.
- [32] 陈黎飞, 姜青山, 王声瑞. 基于层次划分的最佳聚类数确定方法[J]. 软件学报, 2008, 19(1):62-72.
- [33] T. Zhang, R. Ramakrishnan, and M. Livny, “Birch: an efficient data clustering method for very large databases,”in ACM Sigmod Record, vol. 25, no. 2. ACM, 1996, 103-114.
- [34] G. Karypis, E.-H. S. Han, and V. Kumar, “Chameleon: Hierarchical clustering using dynamic modeling,”Computer, no. 8, 68–75, 1999.
- [35] . A. Hartigan and M. A. Wong, “Algorithm as 136: A K-means clustering algorithm,” Journal of the Royal Statistical Society. Series C (Applied Statistics), vol. 28, no. 1, 100–108, 1979.
- [36] R.T.Ng and J.Han,“Clarans:A method for clustering objects for spatial data mining,” IEEE Transactions on Knowledge & Data Engineering, no. 5, 1003–1016, 2002.
- [37] M. Ester, H.-P. Kriegel, J. Sander, X. Xu et al., “A density-based algorithm for discovering clusters in large spatial databases with noise,” in ACM SIGKDD Conference on Knowledge Discovery and Data Mining(KDD), vol. 96, no. 34, 1996, 226–231.
- [38] A. Rodriguez and A. Laio, “Clustering by fast search and find of density peaks,” Science, vol. 344, no. 6191, 1492–1496, 2014.
- [39] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, “Optics: ordering points to identify the clustering structure,” in ACM Sigmod Record, vol. 28, no. 2. ACM, 1999, 49–60.
- [40] Gongde Guo, Hui Wang, David A. Bell, KNN Model-Based Approach in Classification[C]. On the Move to Meaningful Internet Systems: Coopis, Doa, &

Odbase-otm Confederated International Conferences, Coopis, Doa, & Odbase, Catania, Sicily, Italy, November. Springer, Berlin, Heidelberg, 2003.

[41] David Arthur, Sergei Vassilvitskii. K-Means++: The Advantages of Careful Seeding[C]. Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2007, New Orleans, Louisiana, USA, January 7-9, 2007. ACM, 2007.

[42] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi, “Xnor-net: Imagenet classification using binary convolutional neural networks,” in European Conference on Computer Vision. Springer, 2016, 525– 542.

[43] J.-H. Luo, J. Wu, and W. Lin, “Thinet: A filter level pruning method for deep neural network compression,” in Proceedings of the IEEE international conference on computer vision, 2017, 5058–5066.

[44] Y.-D.Kim,E.Park,S.Yoo,T.Choi,L.Yang, and D.Shin,“Compression of deep convolutional neural networks for fast and low power mobile applications,” arXiv preprint arXiv:1511.06530, 2015.

[45] 施诚诚. 基于视频流的弹点捕捉和跟踪[D]. 东华大学, 2014.

[46] 陈雨. 视频车辆检测算法研究[D]. 东北师范大学,2014.

[47] 李武. 多景近景摄影测量的特征匹配,Multiview Close-range Photogrammetry Feature Matching[J]. 北京测绘, 2013, 000(005):46-49.

[48] 李琼. SAR 图像海面油膜提取与分类研究[D]. 中国地质大学（北京）.

[49] 薛婷婷 基于稀疏变换学习的工件视觉跟踪方法研究[D].西安工程大学, 2019

[50] 武洪云, 魏西. 机器视觉技术在防凌破冰中的应用[J]. 无线电通信技术, 2015, 41(1):84-86.

[51] 陈勇涛, 郭晓颖, 陶慧杰. 基于深度学习的图像识别模型研究综述[J]. 电子世界, 2018.

[52] 孙强强. Research on Android Malicious Application Detection Model based on Random Forest. 基于随机森林的 Android 恶意应用检测研究[J]. 信息安全与通信保密, 2019, 000(006):46-54.

攻读学位期间主要研究成果

一、论文

Zhun Fan, Jiewei Lu, Benzhang Qiu, Tao Jiang, Kang An. Automated Steel Bar Counting and Center Localization with Convolutional Neural Networks. Under Review. <https://arxiv.org/abs/1906.00891>

二、专利

[1] 一种基于卷积神经网络的变电站指针式仪表的识别方法 发明人：范衡;安康;姜涛;邱本章;朱贵杰;卞新超;孙福赞 申请号：CN20191143610.6 公开号：CN11092972.3A

[2] 一种手眼触屏机械臂 发明人：范衡;安康;姜涛;邱本章;罗晨林;熊宇 申请号：CN201911075012.X 公开号：CN110900599A

[3] 一种基于卷积神经网络的电厂电表字符定位和识别方法 发明人：范衡;姜涛;安康;邱本章;朱贵杰;王袁野 申请号：CN201910316734.3 公开号：CN110263790A

[4] 一种基于深度卷积神经网络的钢筋端面识别方法 发明人：范衡;卢杰威;邱本章;安康;姜涛;朱贵杰 申请号：CN201811618063.8 公开号：CN109815950A

[5] 一种电厂集控室深度学习和视觉伺服的触屏控制操作方法 发明人：范衡;姜涛;安康;邱本章;罗晨林;熊宇; 申请号：CN201911075065.1 公开号：CN111144401A

致谢

行文最后落笔之处，也意味着研究生三年的学习生涯即将闭幕。时间转瞬即逝，回首三年，百感交集。在这充满生机与学习氛围的汕头大学，留下了青春和沉甸甸的收获，既有万般不舍但仍心怀感激。

在论文完成之际，首先感谢我的指导老师范衡老师和姜永权老师，从本文的选题、资料收集到论文的撰写完成离不开范衡的指导与帮助。他对我的研究提出了很多宝贵的意见，使我写作论文有了目标和方向。范衡老师严谨的治学态度和渊博的学识，朴实无华及平易近人的人格魅力对我影响深远，是我学习的榜样。

其次要感谢我的父母和亲戚朋友，二十余载求学问路，均为父母和亲戚朋友默默付出，在对未来困惑时指点迷津。祝愿你们身体健康并再次感谢你们的无私付出和鼎力支持。

接着要感谢同窗的好友，三年来的朝夕相处你们已经成为我人生中很重要的人，感谢你们的帮助与陪伴，在学习上给予鼓励和支持，在生活上的帮助和开导。愿你们前程似锦，未来可期。最后向此次疫情的逆行者致以最崇高的敬意，幸得有你，山河无恙。祝愿祖国繁荣昌盛国泰民安！

作者：姜涛

2020年5月20日