

硕士学位论文

(专业学位论文)

基于图像处理的血糖监测数据视觉检测

**VISUAL DETECTION OF BLOOD GLUCOSE
MONITORING DATA BASED ON IMAGE
PROCESSING**

张诺敏

哈尔滨工业大学

2021 年 6 月

国内图书分类号：TP391.4

学校代码：10213

国际图书分类号：004.93

密级：公开

硕士学位论文

基于图像处理的血糖监测数据视觉检测

硕 士 研 究 生：张诺敏

导 师：杨旭东 副教授

申 请 学 位：专业学位

学 科：控制工程

所 在 单 位：航天学院

答 辩 日 期：2021 年 6 月

授予学位单位：哈尔滨工业大学

Classified Index: TP391.4

U.D.C: 004.93

Dissertation for the Master Degree in Engineering

**VISUAL DETECTION OF BLOOD GLUCOSE
MONITORING DATA BASED ON IMAGE
PROCESSING**

Candidate:	Zhang Nuomin
Supervisor:	Associate Prof. Yang Xudong
Academic Degree Applied for:	Professional Degree
Speciality:	Control Engineering
Affiliation:	School of Astronautics
Date of Defence:	June, 2021
Degree-Conferring-Institution:	Harbin Institute of Technology

摘 要

对于居家的糖尿病患者来说,将 CGM 屏幕显示的血糖曲线拍照发送给远程管理血糖的医生是目前常用远程医疗方式。考虑到医生需要较为准确的数据进行诊断,因此需要考虑 CGM 血糖图像通过视觉检测进行数据提取。本文针对 CGM 血糖仪屏幕图像设计了数据提取的视觉检测算法,同时开发了 CGM 血糖仪图像数据检测系统。本文的主要研究内容如下:

由于需要针对不同型号 CGM 血糖仪设计不同的视觉检测算法,同时获得大量 CGM 血糖仪屏幕图像比较困难,因此小样本场景下的 CGM 血糖仪屏幕图像分类任务。由于常用的 HOG 特征主要考虑到边缘信息,因此提出改进的 HOG 特征,通过计算图像灰度分布后与 HOG 特征进行拼接,改进后的特征不仅有边缘信息还有图像灰度分布信息。在数据上进行 SVM 分类试验后,得到的结果与原始的 HOG 特征的分类效果又一定提升。

在对 CGM 血糖仪屏幕图像进行数据提取前设计了图像校正环节。校正第一步是通过分割屏幕图像,提出了一种改进 GrabCut 算法,通过区域生长算法计算前景预选框,弥补了 GrabCut 算法需要人工设置预选框的问题。对于分割好的屏幕,采用投影变换进行进一步校正,通过 DP 算法进行边框的四边形拟合,通过拟合后的顶点位置计算投影矩阵,得到了校正后的屏幕图像。

对于屏幕图像,需要提取的数据包括血糖数据与文字数据。血糖数据检测的第一步是检测血糖曲线的坐标轴关键点,通过坐标轴进行进一步校正,获取方向校准好的曲线图像,校准好的曲线图像进行阈值分割,得到了血糖曲线部分,之后进行图像细化,最得到的图像数据进行插值,计算出时间与血糖值的对应关系。对于文本数据的检测,包括文本区域检测与文本分割两步。文本区域检测通过语义分割实现,设计语义分割网络进行文本区域检测,对网络输出进行后处理得到文本区域。文本识别采用 CRNN 网络,包括 CNN 与 LSTM 两部分,对于输出与实际标签不对齐的问题采用 CTC 损失函数,文本区域作为输入放入训练好的 CRNN 网络就可以获得识别出的文字。

最后在设计好视觉检测算法后,开发了 Web 端的 CGM 血糖仪图像数据检测系统。系统采用 Django 框架,包括用户模块、检测模块、管理模块三部分,病人或医生可以通过打开网页端上传图像数据,系统就会调用检测模块自动进行检测,生成 Excel 结构,并保存记录。通过该系统可以辅助医生进行远程医疗。

关键词: 图像处理; 视觉检测; 深度学习; 远程医疗

Abstract

For patients with diabetes at home, it is a commonly used telemedicine method to send the blood glucose curve displayed on the CGM screen to the doctor who remotely manages the blood sugar. Considering that doctors need more accurate data for diagnosis, it is necessary to consider CGM blood glucose image data extraction through visual detection. In this paper, the visual detection algorithm of data extraction is designed for the screen image of CGM blood glucose meter, and the image data detection system of CGM blood glucose meter is developed. The main contents of this paper are as follows

Due to the need to design different visual detection algorithms for different types of CGM blood glucose meter, and it is difficult to obtain a large number of CGM blood glucose meter screen images at the same time, the task of CGM blood glucose meter screen image classification in small sample scenes is very difficult. Because the commonly used hog features mainly consider the edge information, an improved hog feature is proposed. After calculating the gray distribution of the image, it is spliced with the hog feature. The improved feature has not only the edge information but also the gray distribution information of the image. After the SVM classification experiment on the data, the classification results of the original hog features and the results obtained are improved.

Before data extraction of CGM blood glucose monitor screen image, image correction is designed. The first step of correction is to segment the screen image, and an improved grabcut segmentation algorithm is proposed. The foreground preselection box is calculated by region growing algorithm, which makes up for the problem that grabcut algorithm needs to set preselection box manually. For the segmented screen, the projection transformation is used for further correction, the DP algorithm is used for quadrilateral fitting of the frame, and the projection matrix is calculated by the fitting vertex position to get the corrected screen image.

For screen images, the data to be extracted include blood glucose data and text data. The first step of blood glucose data detection is to detect the key points of the coordinate axis of blood glucose curve. Through the further correction of the coordinate axis, the curve image with the direction calibrated is obtained. After the threshold segmentation of the calibrated curve image, the blood glucose curve part is obtained, and then the image is refined. The most obtained image data is interpolated to calculate the corresponding relationship between time and blood glucose value. For text data detection, including text region detection and text segmentation. The text region detection is realized by semantic segmentation. A semantic segmentation network is designed to detect the text region, and the output of the network is processed to get the text region. Text recognition uses crnn network, including CNN and LSTM. For the problem that the output is not aligned with the actual label, CTC loss function is used.

The text area is put into the trained crnn network as the input to get the recognized text.

Finally, after designing the visual detection algorithm, the CGM blood glucose meter image data detection system on the web is developed. The system uses Django framework, including user module, detection module and management module. Patients or doctors can upload image data by opening the web page. The system will call the detection module to automatically detect, generate excel structure and save records. The system can assist doctors in telemedicine.

Keywords: image processing, visual inspection, deep learning, telemedicine

目 录

摘 要.....	I
Abstract.....	II
目 录.....	IV
第一章 绪论.....	1
1.1 课题背景及意义.....	1
1.2 国内外研究现状.....	3
1.2.1 视觉检测的传统算法.....	3
1.2.2 视觉检测的机器学习算法.....	4
1.2.3 视觉检测的深度学习算法.....	5
1.3 本文主要研究内容和结构安排.....	7
第二章 小样本图像分类研究.....	9
2.1 引言.....	9
2.2 基于 HOG 特征的小样本识别.....	9
2.2.1 HOG 特征计算.....	9
2.2.2 数据集制作.....	12
2.3 支持向量机分类器介绍.....	13
2.4 HOG 特征的 SVM 分类实验.....	15
2.5 改进 HOG 特征的 SVM 分类实验.....	16
2.6 本章小结.....	19
第三章 图像校正.....	20
3.1 引言.....	20
3.2 分割屏幕图像.....	20
3.2.1 区域面积重采样插值.....	20
3.2.2 GrabCut 分割原理.....	21
3.2.3 改进 GrabCut 分割.....	23
3.3 投影变换校正.....	26
3.3.1 投影变换原理.....	26
3.3.2 轮廓拟合.....	27
3.3.3 投影变换校正.....	29
3.4 校正算法实验验证.....	30

3.5 本章小结.....	32
第四章 图像数据视觉检测.....	33
4.1 引言.....	33
4.2 血糖数据视觉检测.....	33
4.2.1 检测坐标系与血糖曲线分割.....	33
4.2.2 图像细化与血糖值计算.....	37
4.2.3 算法实验验证.....	39
4.3 文字数据检测.....	40
4.3.1 文本检测的语义分割网络设计.....	40
4.3.2 文本区域检测的训练与后处理.....	43
4.3.3 CRNN 文字识别.....	45
4.3.4 算法实验验证.....	48
4.4 本章小结.....	48
第五章 CGM 血糖仪数据检测系统开发.....	50
5.1 引言.....	50
5.2 总体架构与需求分析.....	50
5.2.1 系统总体架构.....	50
5.2.2 系统需求分析.....	51
5.3 系统模块开发.....	52
5.3.1 功能模块设计.....	52
5.3.2 功能模块实现与测试.....	54
5.4 本章小结.....	58
结论.....	59
参考文献.....	61
攻读硕士学位期间发表的论文及其它成果.....	64
哈尔滨工业大学学位论文原创性声明和使用权限.....	65
致谢.....	66

第一章 绪论

1.1 课题背景及意义

糖尿病是一种常见的慢性代谢性疾病，主要表现为体内胰岛素分泌不足或胰岛素反应缺陷引起的持续性高血糖^[1]。糖尿病的类型包括 1 型糖尿病、2 型糖尿病、妊娠期糖尿病以及其他特殊类型糖尿病。糖尿病对病人的危害巨大，无论是哪种类型的糖尿病，如果患者不能及时得到有效治疗，血糖浓度将长期维持在较高水平，会进一步引发严重的糖尿病并发症，这也是糖尿病患者致残、致死的主要原因。常见的糖尿病并发症包括：大血管病变、肾脏病变、眼部视网膜病变、糖尿病足等^[2]，并发症会进一步导致对其他人体脏器造成严重损害，严重危害患者身体健康。

糖尿病作为一种常见的慢性非传染性疾病，已成为严重的世界性公共卫生问题。国际糖尿病联盟(IDF)发布的数据显示，预计到 2045 年，全球糖尿病患者总数将增至 6.29 亿。随着我国经济发展、城市化进程的加快以及人口老龄化的加剧，我国糖尿病等慢性非传染性疾病的发病率将大幅上升：我国糖尿病患病率从低于 1%迅速增长至超过 10%仅用了 30 年，且在一段时间内仍将呈现增加趋势^[3]。2019 年 6 月发布的《国务院关于实施健康中国行动的意见》指出，我国是糖尿病患病率增长最快的国家之一。根据最新的流调数据以及世界卫生组织的诊断、标准作为依据参考，我国 2015 至 2017 年的糖尿病发病率已达到 11.2%，相较 2013 年至 2014 年的 10.4%，和 2007 年至 2012 年的 9.7%有明显提升。随着我国人民生活水平的提高，城镇化程度的提高以及人口老龄化加剧，糖尿病的患病率还会进一步增加而不会降低。由此而带来糖尿病本身的危害及其并发症的高致残致死率，不仅会带来严重的健康问题，而且增加医疗费用，加重医疗卫生与社会经济的负担，将会成为我国公共卫生事业最为严峻的挑战。

随着社会经济的不断发展，网络设施的建设逐年完善，我国互联网的普及率也逐年上升远程医疗技术新模式，已逐步成为糖尿病慢性疾病管理系统的有效载体^[4]。远程医疗通过采用互联网与移动通信技术实现远程通信，能够及时查看患者的健康状况，并适时向病人反馈治疗建议，帮助患者在居家时获得治疗改善健康，实现了包括诊断、治疗与监护在内的远程医疗活动^[5]。糖尿病作为一种慢性疾病，大部分患者都是长期居家治疗。同时我国患者数量巨大，为医护人员带来巨大压力。因此发展基于互联网技术的远程医疗服务对加强慢性病患者健康管理和改善

医患矛盾具有重要意义^[6]。

糖尿病的治疗是一个综合管理的过程，需要饮食、运动、药物、血糖监测和健康教育同时推进。如果对患者治疗不及时，血糖波动会造成机体器官的损害，加速并发症的发生。因此血糖监测是糖尿病治疗中的重要一环。动态血糖监测(Continuous Glucose Monitor, CGM)在血糖监测指南中被列为血糖监测方法之一。CGM 是一种便携的血糖监测医疗仪器，包括传感器、发射器和接收器三部分构成，由皮下传感器实现实时测量皮下组织间液中的葡萄糖水平。与传统的自我血糖监测方式相比，CGM 不但能够避免手动测量产生的测量误差的影响，而且能够提供连续、全面、可靠的全天血糖信息，了解血糖波动趋势，发现不易被传统方法发现的隐匿性高血糖和低血糖，具有良好的准确性和安全性，使患者更好地了解自己的血糖变化情况，包括空腹、餐后及运动后等情况下血糖的波动情况，从而降低血糖异常的发生率^[7]。经过一系列的临床实验后，可以证明通过使用 CGM 及时查看患者血糖数据，能够及时发现血糖异常状态，有利于调控血糖水平，保证患者血糖较长时间为正常状态，进而能够降低血糖异常的风险。



图 1-1 CGM 血糖仪图像

目前对于居家治疗的糖尿病患者来说，对于为获得足够的医护人员的帮助，病人将 CGM 血糖仪屏幕显示的血糖曲线拍照发送给远程管理血糖的医生是一种常用的方式。这是因为目前开发的 CGM 血糖仪的数据虽然可以提取，但是操作比较复杂，并且需要连接电脑，不方便老年人使用。虽然医生看到照片后可以对系统曲线数值进行估算，但并不准确，而医生对于血糖异常分析的实时性较高，至少当天需要知道血糖波动原因并给出处量意见，并且胰岛素剂量的计算需要依赖于准确的血糖数据，肉眼的估算不能准确反映患者真实血糖水平。所以，目前需要一个可以帮助医生量化的软件，并帮助医生对多位病人数据进行管理与记录。

1.2 国内外研究现状

1.2.1 视觉检测的传统算法

视觉检测是计算机视觉技术的一个重要应用领域,是指通过图像处理软件对给定图像进行处理,将这些图像数字信号执行相应的数学运算以提取特征信息,例如:长度、面积、形状、位置等;最后再根据预先设定的判断条件输出相应的处理结果,来指导实际活动,例如:合格/不合格、精度等级、数量等。通常视觉检测算法的流程包括预处理、特征提取、处理特征提取数据等步骤。

早期的视觉检测算法主要是通过人工设计滤波器进行图像滤波去噪;基于canny、sobel等算子的边缘检测方法以及小波变换、频率域法等来实现目标特征的提取、区域分割、边缘检测、角点检测;基于灰度的大津算法、漫水填充法、分水岭算法等的目标分割。此时检测的场景比较简单,待检测目标在纹理,颜色,灰度,形状等特征比较明显,适合将灰度、边缘、颜色等图像的直观特点作为待检测的特征,并通过阈值分割等方法来完成目标的识别。赵亚湘通过方向滤波器实现了对指纹图像的去噪与增强,提高处理效果^[8]。冯莉等人在原木材积视觉检测系统中,根据原木外廓形状的具体特点,将边缘信息作为特征,采用二值化方法分割原木图像,最终获得原木测量信息^[9]。林盘等人在指针式仪表视觉检测中,使用中值滤波对于指针式仪表图像波进行预处理去噪,并用矩不变自动门限图像分割算法分割指针,实现了仪表测量^[10]。常辉在套件组装视觉检测中,为了解决光照不均匀的问题,采用直方图均衡化对产品图像进行处理,并用一维小波变换提取图像边沿上的犄角点,并利用犄角点之间的联系实现对套件产品的识别^[11]。王鹏在钢球表面缺陷检测系统中,对于动态采集的图像,使用Canny算子进行边缘检测,进而通过边缘特征实现图像的分割,获得想要提取的目标的信息,对得到的结果采用标准钢球图像生成进行模板进行模板匹配,模板由标准钢球图像生成,实现了对钢球的裂纹、裂缝的检测^[12]。苗世迪在机械零件尺寸计算机视觉测量中,使用中值滤波方法进行预处理,之后对圆的Hough变换进行改进,同时边缘检测方法进行了初步的研究,采用多项式插值亚像素的边缘检测,开发出适合工业上的机械零部件尺寸测量的视觉检测算法^[13]。

对于传统算法而言,使用这些技术的思路是:从统一类别对象的图像中提取尽可能多的特征,并将这些特征视为对象的一个“定义”,然后在其他图像中搜索这些“定义”。这种特征提取方法的难点在于,在给定图像中,必须人工选择需要查找哪些特征。当图像中类别过多时,挑选得特征就会变得复杂而难以分类。而只

有使用不同的特征才可以更好地描述不同类别的对象。如果你在分类时使用很多特征，就必须对大量的参数进行微调。因此传统视觉检测算法由于其人工提取特征和场景指定性强的原因，存在着模型泛化能力不强和鲁棒性差的问题，在图像中噪声存在和背景以及目标类别改变的情况下性能会受到较大影响。

1.2.2 视觉检测的机器学习算法

随着机器学习的发展，许多机器学习算法包括统计学习算法、感知机、主成分分析、KNN、支持向量机、XGBoost、聚类算法等机器学习算法也在视觉检测领域得到广泛应用。闵莉等人提出了一种基于支持向量机算法的红外图像目标检测，首先通过选择合适的阈值和有效的边缘检测算子对候选目标图像进行二值化和边缘检测，之后以候选目标的边界为不变矩特征，使用支持向量机方法进行目标识别，能实现对红外目标有效的检测与识别，并具有较高的抗噪声能力，同时提高了在复杂背景下的效果^[14]。李靖等人提出了白化主成分分析类算法的框架，该算法框架的主要思路是使用白化滤波器和低通滤波器对输入图像进行处理，对得到的结果使用船艇的 PCA 类算法实现进一步提取特征向量，最后使用 KNN 方法实现最终的人脸识别^[15]。杨灿通过利用 XGBoost 算法从遥感图像中的提取道路部分，以图像中像素点的颜色特征和局部特征作为特征输入，进而实现对图像中的每个像素点的分类，最终实现了遥感图像中道路网络的提取^[16]。胥海威提出的一种基于改进随机聚类决策森林算法的遥感影像分类算法中对随机聚类决策森林算法进行了相关改进包括添加树平衡系数；对叶节点的节点不纯度予以调整；结合随机聚类决策森林算法特点，采用决策预剪枝技术进行改进等，并在遥感图像分类任务中使用改进的算法，取得了不错的结果^[17]。翁秉钧等人提出一种基于改进 K 均值聚类的输电线路覆冰状态检测方法，在获取输电线路图像后，进行预处理，之后依次使用直线段检测算法、改进 K 均值聚类、最小二乘拟合确定导线位置，计算导线宽度，并根据覆冰前后宽度变化评估其覆冰状态^[18]。

机器学习方法在视觉检测领域相较于传统视觉处理方法有较好的模型泛化能力，通过模型学习让模型自主学习图像和目标特征，增加了模型的鲁棒性。但是在深度学习技术大范围应用之前，传统机器学习在视觉检测中应用很大程度还是靠人工设计特征，例如主成分分析、尺度不变特征变换(Scale-invariant feature transform, SIFT)、加速稳健特征(Speeded up robust features, SURF)、方向梯度直方图(Histogram of oriented gradients, HOG)以及支持向量机核函数等，依赖于经验去选取，并且严重影响模型效果。同时对于大规模、多维度、多类别的类型数据，

传统机器学习方法由于其模型表达能力的限制无法有好的性能表现。

1.2.3 视觉检测的深度学习算法

1986 年, Hinton 提出了反向传播算法(BackPropagation, BP)。BP 算法在神经网络正向传播之后, 增加了误差的反向传播过程。通过反向传播过程, 求出误差对各个参数的偏导数, 进而可以通过梯度下降不断调整神经元之间的各项参数, 直到模型在数据集上的误差减小到给定范围或达到预定的训练时间^[19]。但是由于八十年代硬件的计算能力有限, 再加上 90 年代中期, 以支持向量机为代表的其它机器学习算法被提出, 并在分类和回归等方面取得了较好的效果, 其原理又明显不同于神经网络模型, 具备较强的解释性, 所以神经网络的研究进入了瓶颈期。

卷积神经网络(Convolutional Neural Networks, CNN)是一种在计算机视觉中常用的神经网络模型, 是深度学习的代表算法之一, 与普通的神经网络相比, 主要特点是隐含层的卷积层和池化层是特征提取功能的核心模块。1998 年 Yann LeCun 及其合作者研究设计了卷积神经网络 LeNet-5, 虽然网络很小, 但是它包含了深度学习的基本模块: 卷积层, 池化层, 全连接层等结果, 因此这是最早的比较完备的卷积神经网络, 并且在手写体数字识别中取得成功^[20]。LeNet5 通过巧妙的设计, 利用卷积、参数共享、池化等操作提取特征, 很好利用图像的结构信息并减少待学习的参数个数, 避免了大量的计算成本, 因此这个网络也是最近大型卷积神经网络架构的起点。2012 年, 在著名的 ImageNet 图像识别大赛中, 深度学习模型 AlexNet 一举夺冠。AlexNet 采用 ReLU 激活函数, 从根本上解决了梯度消失问题, 并采用 GPU 极大的提高了模型的运算速度^[21]。在此之后进入了深度学习的爆发阶段, CNN 成为图像领域特征提取最好的方式, 深度学习开始在图像分类、目标检测以及其他领域中广泛应用。

在 AlexNet 之后, 又出现许多有代表意义的深度学习模型。2014 年, 牛津大学和 Google 的研究员一起研究设计出了新的卷积神经网络 VGGNet, 并取得了 ImageNet 比赛分类项目的第二名。VGGNet 通过反复的堆叠 3*3 的小型卷积核和 2*2 的最大池化层, 成功的构建了 16~19 层深的卷积神经网络, 同时探讨了卷积神经网络的深度与其性能之间的关系^[22]。同年 2014 年 Christian Szegedy 提出的一种全新的深度学习结构 GoogLeNet, 在 2014 年的 ImageNet 竞赛中夺得了冠军。GoogLeNet 通过 inception 模块, 避免通过直接增大网络的深度带来的负作用, 比如过拟合、梯度消失、梯度爆炸等^[23]。2015 年, He, K 等人提出 Resnet, 并在 2015 年 ImageNet 比赛中获得了冠军。Resnet 使用 ResNet 引入了残差网络结构, 通过

Identity mapping 的引入在输入、输出之间建立了一条直接的关联通道，从而使得强大的有参层集中精力学习输入、输出之间的残差^[24]。通过该方式，ResNet 解决了深度卷积神经网络模型难训练的问题。到目前为止，神经网络的层数可以超过以前的限制，达到几十层、几百层甚至上千层，进而可以实现高级语义特征的提取，因此在工业界中 Resnet 也成为常用的基础特征提取网络，同时残差网络结构在深度学习网络设计时被广泛应用。

在此之后目标检测、语义分割等领域广泛使用卷积神经网络作为基础特征提取网络，取得了很大的突破。目标检测算法包括分别为 One-Stage 目标检测算法和 Two-Stage 目标检测算法。Two-Stage 目标检测算法一般是先算法产生目标候选框，也就是目标位置，然后再对候选框做分类与回归，主要有 Faster R-CNN、Cascade R-CNN 等算法。One-Stage 算法早期以有锚框的检测算法为主，包括 SSD、YOLO 系列目标检测算法，后来又有无锚框的目标检测算法，以 CenterNet 为代表。与 Two-Stage 相比，One-Stage 算法直接在输出层回归预测框的位置和类比，运行速度更快。语义分割是对图像中的每一个像素进行分类，目前广泛应用于医学图像与无人驾驶等。2015 年 Jonathan Long 等人提出全卷积神经网络 FCN，去掉了卷积神经网络中常用的全连接层，而是使用端对端卷积神经网络，并使用反卷积进行上采样，为之后语义分割工作提供基础^[25]。2015 年 U-net 网络的提出使得语义分割在医学图像领域受到广泛应用。U-net 网络使用编码器-解码器架构，在医学图像中获得广泛应用^[26]。Chen 等人提出 DeepLab 系列语义分割网络，使用空洞卷积以及 SPP 结构，利用多种比例和多种有效感受野的不同分辨率特征处理，来挖掘多尺度的上下文内容信息，并使用编码器-解码器结构逐步重构空间信息来更好的捕捉物体边界^[27]。

与传统的机器学习算法相比，深度学习在大数据集上可以使用更多的数据，扩展性更好。同时，传统的机器学习算法通常需要复杂的特征工程来选择任务中的最佳特征；而当使用深度学习时，不需要人工设计特征，只需将数据直接送到网络进行训练，神经网络就可以自己学习提取特征，就可以达到不错的性能。在数据集比较大时，这完全消除了整个过程中庞大而富有挑战性的特性工程阶段。并且深度学习技术可以更容易地在不同的领域和应用进行泛化。

然而，目前深度学习还存在一些问题：深度学习网络的结构设计缺乏统一的原则，目前学术界和工业界有多种深度学习模型。各种深度学习模型的收敛速度无法定量设计，重要优化算法的性能需进一步优化。为了实现高性能，深度学习需要非常大的数据集，对于许多应用来说，这样的大数据集并不容易获得，并且花费昂贵且耗时。而深层神经网络训练时也需要 GPU 的高算力进行训练。在实际

应用中，部署一个深度学习模型和算法非常困难，需要有相当多的工程经验和技巧，这些也是深度学习的劣势。

1.3 本文主要研究内容和结构安排

本论文主要研究 CGM 血糖仪屏幕图像的视觉检测，针对 CGM 血糖仪屏幕图像设计了数据提取的视觉检测算法。在数据提取中，由于对于不同类型的图像，检测算法的参数以及方式会有不同，因此需要进行分类算法研究，对输出图像识别出种类，进而确定后续的检测算法。而为了提高检测的准确率，在算法检测之前加入图像校正环节。因此整个视觉检测算法包括图像分类、图像校正、提取数据三个部分。

在研究视觉检测算法之后，还实现了实现 CGM 曲线识别与数据检测系统开发，方便病人上传数据并进行自动检测数据，同时可以帮助医生保存并管理病人数据，辅助医生进行远程医疗。因此整个检测系统工作的流程图如下：

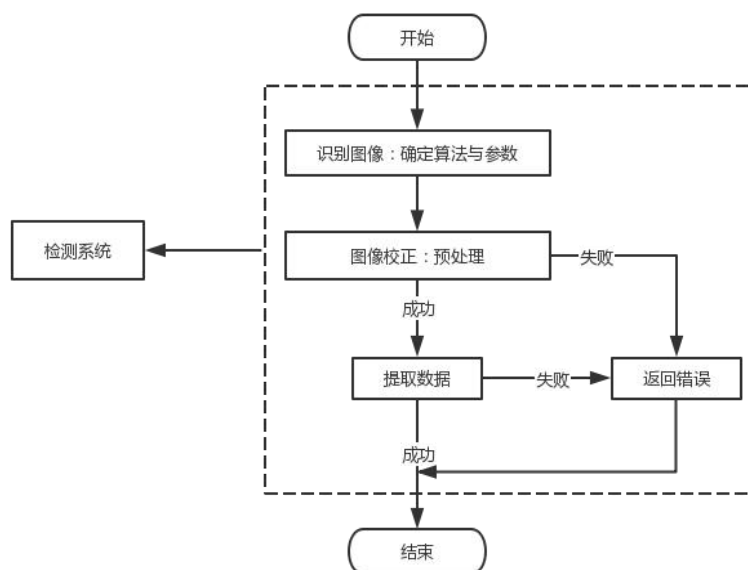


图 1-2 视觉检测算法流程图

根据以上的设计思路，本文共分五章，每章的主要内容如下：

第一章为绪论部分。首先为糖尿病背景介绍，说明了糖尿病在中国的严峻形势，并介绍 CGM 血糖仪等在远程医疗的糖尿病控制中应用，同时介绍 CGM 血糖仪屏幕图像的视觉检测的意义，然后总结了视觉检测算法的应用，包括传统检测方法以及后来发展的机器学习方法与深度学习方法，最后介绍了本文的主要研究内容和论文结构。

第二章对 CGM 血糖仪屏幕图像的分类算法进行了研究。在针对小样本的情况

下, 研究基于 HOG 特征与 SVM 的分类算法在 CGM 血糖仪屏幕图像的分类上的应用, 同时根据提出改进 HOG 特征方法, 并进行实验结果对比, 研究出合适的分类算法。

第三章针对 CGM 血糖仪屏幕图像的校正算法研究, 这里提出一种基于区域生长算法计算预选框的改进的 Grabcut 算法, 并验证该算法适用于分割具有明显与背景不同边框的图像。在分割屏幕图像之后使用投影变换矫正血糖仪屏幕图像, 为下一步提取图像信息做准备。

第四章为数据提取算法研究, 包括两类数据, 一是血糖曲线数据, 另一种是时间与日期数据等。通过检测坐标直线与灰度值等特征, 分割出血糖数据曲线部分, 使用图像细化算法提取血糖数据曲线, 并通过插值算法获得与时刻对应的血糖数据。对于日期等数据, 使用语义分割算法获取图像文字区域, 并使用 CRNN 模型获取文字信息。

第五章为 CGM 曲线识别与数据检测系统开发, 使用 Django 框架开发出 CGM 图像视觉检测系统, 满足医生的需求, 帮助医生对糖尿病患者进行远程血糖管理。

第二章 小样本图像分类研究

2.1 引言

本章任务主要是通过小样本的数据进行 CGM 血糖仪屏幕图像进行分类。由于当前市场上 CGM 血糖仪有多种型号,在后续数据提取中,针对不同型号 CGM 血糖仪,需要设计不同的视觉检测算法。因此视觉检测的第一步是将待检测图像进行分类,进而确定后续的检测算法。而该分类任务与常见的图像分类的区别在于,目前无法获得大量的 CGM 血糖仪屏幕图像,因此本章主要考虑在小样本场景下的 CGM 血糖仪屏幕图像分类任务。

2.2 基于 HOG 特征的小样本识别

方向梯度直方图(Histogram of oriented gradients, HOG)提出于 2005 年,是在计算机视觉领域中常用的特征描述子。该特征描述子通过计算图像局部区域的梯度方向直方图来构造特征。目前对图像提取 HOG 特征并结合 SVM 分类器的方法在目标检测与图像识别中被广泛使用,特别是在行人检测方面取得了很大的成功^[28]。通过 HOG 特征对图像进行分类的主要思路是先计算图像的 HOG 特征,并使用软间隔支持向量机对提取的 HOG 特征进行分类,进而实现图像的分类。

2.2.1 HOG 特征计算

在一副图像中,图像的表象与特征主要体现为边缘信息,这能够被梯度的方向密度分布很好地描述,因此 HOG 主要靠统计图像的梯度生成特征。

在计算之前需要对图像进行预处理,主要包括灰度化与 Gamma 变换矫正两部。对图像进行灰度化的原因在于 HOG 提取的是边缘特征,图像的颜色信息不作为主要特征,所以将 RGB 三通道图转化为灰度图。对转换后的灰度图像进行 Gamma 变换预处理。Gamma 变换公式为:

$$I(x, y) \leftarrow I(x, y)^\gamma \quad \gamma = 0.5 \quad (2-1)$$

通过非线性变换预处理,图像对比度降低,降低局部光照不均匀或者阴影的影响,同时也可以在一定程度上降低噪声以及光照等干扰因素的影响,提高特征描述器对的鲁棒性。

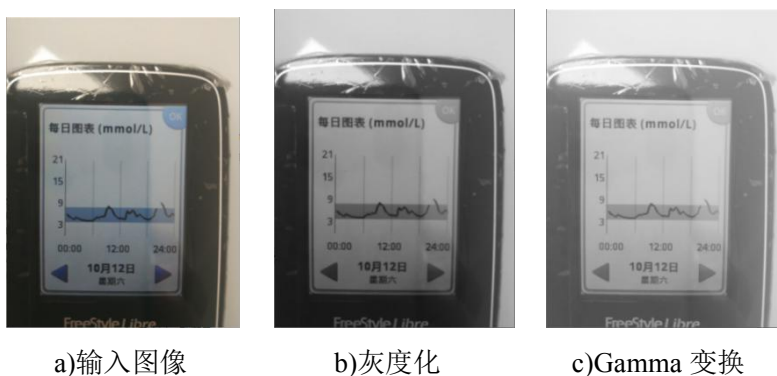


图 2-1 HOG 预处理

预处理之后先计算图像的梯度。梯度使用 Sobel 算子进行计算，包括水平与垂直两个方向的梯度。Sobel 算子垂直方向和水平方向的模板如下图，将之与图形进行平面卷积，就可以得到图像横向及纵向的梯度。

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

a)垂直方向模板 b)水平方向模板

图 2-2 Sobel 算子模板

然后计算每个像素点的梯度幅值和方向。这里梯度方向将会取绝对值，因此梯度方向的范围是 0-180 度。

$$g_x = L(x+1, y) - L(x-1, y) \quad (2-2)$$

$$g_y = L(x, y+1) - L(x, y-1) \quad (2-3)$$

$$\theta = \arctan(g_y / g_x) \quad (2-4)$$

水平方向梯度图会强化垂直方向的特征，垂直方向梯度图会强化水平方向特征。因此通过计算梯度移除了大量非显著性特征，并加强了显著特征，而且可以弱化光照不均匀的影响。

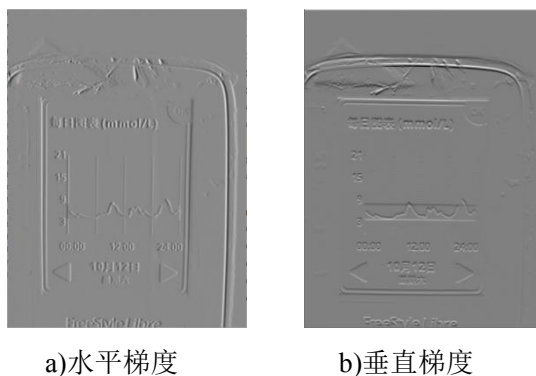


图 2-3 梯度图

经过上一步计算之后，每个像素点都会有两个值：梯度方向和梯度幅度。梯度幅度和梯度方向图与原图等同大小，但是如果直接使用这些特征，会存在两个问题特征是稀疏的。观察计算得到的梯度可以看到，图像中其实只有少量稀疏的显著特征，因为大部分梯度数值在零值附近。HOG 特征在这里将图像分成一系列 cell 单元格，每个 cell 由 8×8 个像素组成，然后在每个 cell 内对梯度幅度和梯度方向进行统计获得对应的直方图。统计方法为将 $0-180^\circ$ 分为若干个区间，这里选择分成 9 个区间。然后统计以梯度为权重总计每个区间梯度，这样每个 cell 就可以获得一个 9 维向量。例如下图：某个位置的梯度方向为 178，幅值为 75，则需要在向量的最后一个维度加上 75，这样就得到了每个 cell 内的特征向量，整个更多的点的梯度方向是倾向于 0-20 和 160-180，表明图像这个位置存在比较明显的横向边缘。通过这样的统计方法，大大降低了计算量，同时又使得计算出的特征对图像灰度变化不敏感，所以能够适应不同的环境。

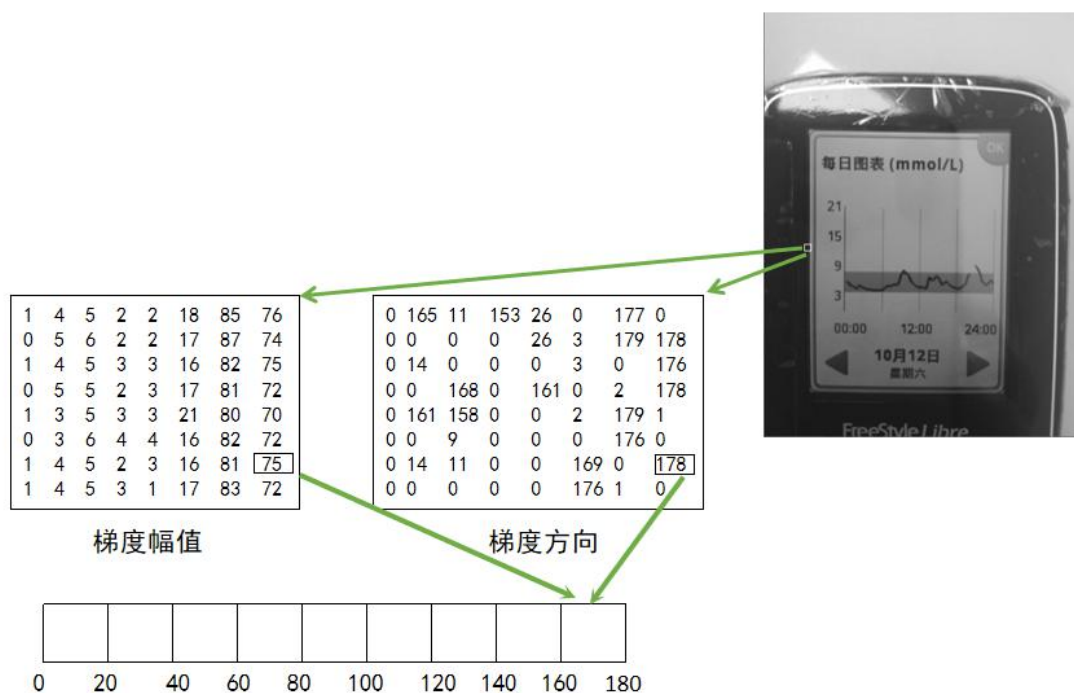


图 2-4 统计 cell 内梯度

在统计 cell 内梯度后，再以 4×4 个 cell 为一组，称为 block。对于一个 block，把所有包含的 cell 的得到的特征向量拼接便得到该 block 的特征向量。因此每个 block 得到数据为 $4 \times 4 \times 9$ 个，也就是 144 维向量。每一个 block 以固定步长方式进行滑动重复计算，直到整个图像的 block 都计算完成。由于局部光照的不均匀，以及前景背景对比度的变化，导致梯度的变化范围非常大，因此有必要对每个 block

的特征向量做归一化。这里使用 L2 范数进行归一化。 ε 其中为一个很小的数，加入 ε 是为了防止分母为 0。

$$v \leftarrow \frac{v}{\|v\|_2 + \varepsilon} \quad (2-5)$$

通过这样的方式最终得到了整个图像的 HOG 特征。我们这里计算 HOG 特征时，每个 cell 为 8×8 ，每个 block 为 32×32 ，block 的滑动步长为 16，输入图像为 512×384 ，因此这里需要计算 block 个数 n 为：

$$n = \left(\frac{512 - 32}{16} + 1\right) \times \left(\frac{384 - 32}{16} + 1\right) \quad (2-6)$$

所以最后整幅图像得到的 HOG 特征向量为 $n \times 144$ ，也就是 102672 维向量。对于该向量，我们就可以作为支持向量机的输入进行分类。

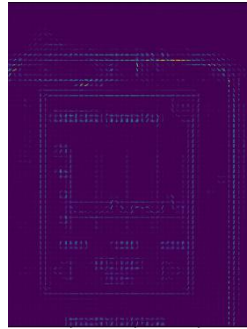


图 2-5 计算得到的 HOG 特征

2.2.2 数据集制作

数据集主要通过拍摄 CGM 血糖仪屏幕图像进行采集。这里采集到的 CGM 血糖仪屏幕图像主要包括 24 小时与 8 小时两种。这里将采集到的图像分成训练集与测试集两类。训练集包括 7 张 24 小时图像与 4 张 8 小时图像，测试集包括 10 张 24 小时图像与 10 张 8 小时图像。

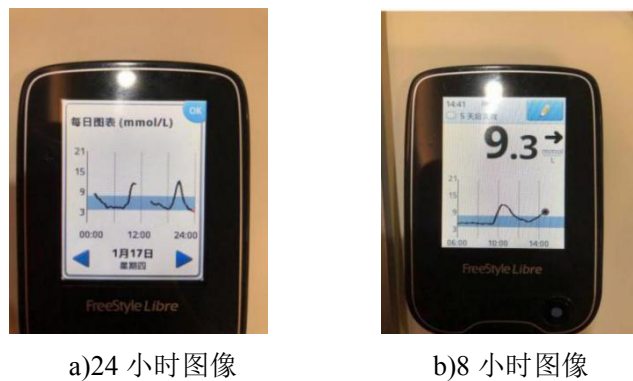


图 2-6 图像种类

但是考虑到训练集数据过少,为了处理该问题,本文使用了数据增广方法。数据增广可以在有限的数据集上产生更多的数据,进而可以增加训练样板数据的数量以及多样性,同时在数据中增加噪声,进而避免过拟合,使模型拥有更好的泛化能力。查看对比 Omniglot 和 MiniImage 等小样本学习常用数据集,发现每一个类别均包含 20 个样例左右。因此针对数据量在小于 20 的类别,本文将通过旋转、缩放、高斯噪声等数据增强方法将训练集数据量扩充为 20。

2.3 支持向量机分类器介绍

在按照上一节通过提取目标识别特征之后,需要使用分类器对特征向量进行处理,输出目标类别,这里使用支持向量机(Support Vector Machine, SVM)作为目标类别的分类器^[29]。

支持向量机常用于监督学习中的二分类问题,核心思想是寻找一个分类器使得超平面和最近的数据点之间的间距最大,对于 SVM 算法通常认为数据与分类超平面距离越大,得到的分类平面越优,因此定义具有最大间隔的决策面就是 SVM 要寻找的最优解,并且最优解对应两侧离分类平面最近的数据点为支持向量。^[30]

(1)线性可分 SVM:

以最简单的二分类说明,设数据集为:

$$D=\{(x_i, y_i) | x_i \in R^p, y_i \in \{-1, 1\}\}_{i=1}^n \quad (2-7)$$

用函数 (2-8)表示,当 $f(x) > 0$ 时,输入 x_i 赋值为正类;当 $f(x) < 0$ 时,输入 x_i 赋值为负类。超平面可表示为式 (2-8):

$$f(x) = \omega^T x + b \quad (2-8)$$

当 $f(x) > 0$ 时,对应 $y=1$ 的正类数据;当 $f(x) < 0$ 时,对应 $y=-1$ 的负类数据;当 $f(x)=0$ 时, x 是位于超平面上点。因此对数据点 x 进行分类时,只需判定函数 $f(x)$ 的正负号,如果 $f(x) > 0$, 则赋值 1, 如果 $f(x) < 0$, 则赋值-1。

在对数据得到分类结果后,分类结果的确信度取决于该数据点到分类超平面的距离,距离越大,证明该数据点到分类超平面越远,那么分类结果的确信度越高。为了提高分类的可信度,需要通过优化点到分类超平面的距离来寻找最优的分类超平面,使分类器具有最佳的性能和更好的泛化能力^[31]。

设 H 为超平面, H_1 和 H_2 为平行于 H 的两个平面并且满足距离样本最近。 H 与 H_1 和 H_2 距离为:

$$\delta = \frac{2}{\|\omega\|} \quad (2-9)$$

SVM 通过求解最大几何间隔获取分类的超平面，因此 SVM 的优化目标为为：

$$\max_{w,b} \frac{2}{\|w\|} \Rightarrow \min_{w,b} \frac{1}{2} \|w\|^2 \quad s.t. \quad y_i(w^T x_i + b) \geq 1 \quad (2-10)$$

该问题可首先使用拉格朗日乘子法，把所有的约束条件，分别乘上拉格朗日乘子，添加到需要优化的目标函数里，形成一个待优化的表达式。对于该表达式，求解对应的对偶问题，进而获得最终的分类模型。

(2) 软间隔 SVM

在实际过程中，很多问题不是线性可分的问题。为了解决这些问题，引入了软间隔，即允许一些样本点跨越间隔边界甚至是超平面，例如下图中一些离群点就跨过了间隔边界。

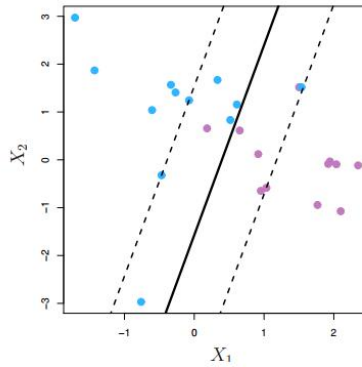


图 2-7 软间隔 SVM

通过软间隔的计算方式，为每个样本点引入松弛变量 ξ_i ，放松了支持向量机的基本约束条件的限制，并在原来的目标函数上增加一项惩罚函数。从而该情况下 SVM 的优化问题变为：

$$\min_{w,b,\xi} \frac{1}{2} \|w\|^2 + C \sum_{i=1}^m \xi_i \quad s.t. \quad y_i(w^T x_i + b) \geq 1 - \xi_i \quad \xi_i \geq 0 \quad (2-11)$$

由上式可以看出：

- a) 离群点的松弛变量值越大，点就离间隔边界越远。
- b) 所有没离群的点松弛变量都等于 0，即这些点都满足 $y_i(w^T x_i + b) \geq 1$ 。

c) C 被称为惩罚参数，若 C 值太大，被错误分类的样本对分类器影响大，不利于分类器的泛化；C 值太小，被错误分类的样本对分类器影响小，分类器的性能可能不佳。因此 C 的取值需要根据实际运用场景进行权衡，

优化问题的求解方法与硬间隔 SVM 类似，使用拉格朗日乘子法，为每条约束引入拉格朗日乘子之后，计算其对偶问题。

2.4 HOG 特征的 SVM 分类实验

对于 CGM 血糖仪屏幕图像进行分类识别，本节采取 HOG+SVM 模型，采用 HOG 特征描述子为特征提取器，采用 SVM 为分类器，从而实现图像的分类。分类模型的工作流程主要分为两个过程：训练过程和测试过程，下面对其简要说明。

(1)训练过程：输入训练集样本图像到模型，使用 HOG 算法提取该图像的边缘特征，则得到该图像的特征向量，作为特征向量送入 SVM 分类器，SVM 分类器输出目标类别，将输出类别与实际类别进行比较，不断调整 SVM 的分类参数，使 SVM 分类器能将样本正确分类，直到所有训练集图像数据训练完，得到 SVM 分类器的最优分类参数，此时 SVM 分类器可对未知类别的测试图像进行分类识别。

(2)测试过程：设计的 SVM 分类模型经过训练集图像数据训练后，模型的分类参数确定，即可对未知类别的测试图像进行分类识别，使用测试集图像输入到 SVM 分类模型，则输出目标类别，根据测试结果对分类模型的工作性能进行评估。

完整的分类模型的训练与测试的流程图如下：

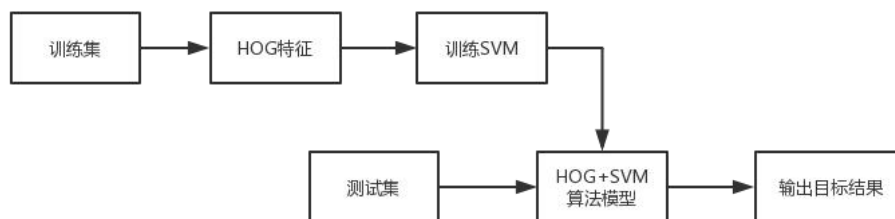


图 2-8 HOG+SVM 分类

对于模型分类的评估，这里采用混淆矩阵作为评估指标。混淆矩阵是分类精度评估的标准格式，是一个 $n \times n$ 的矩阵。在图像分类中，可以统计模型预测分类结果并和实际值进行对比，从而评估分类效果，因此可以通过获取混淆矩阵评估模型分类的效果。混淆矩阵的每一列数据代表算法分类的结果，每一列的数据的总和代表数据集中被分类为对应类别的样本的数量；每一行数据的数据集中真实类别，每一行的数据总数代表数据集中对应类别的样本的数目。通过混淆矩阵统计的数目，就可以计算分类各类的准确率^[32]。

表 2-1 混淆矩阵

Confusion Matrix		分类值	
		Class 0	Class 1
真实值	Class 0		
	Class 1		

分类模型的软硬件环境为：windows10，I5-10400，2.90G HZ，Python3.6。算法模型训练与测试的基本流程为：

①获取训练集和测试集样本，24 小时 CGM 图标签为 0，8 小时 CGM 图标签为 1。

②将每张图像大小统一调整为 384x512，这是因为目前手机拍摄图像大多为 4:3。

③使用 HOG 算法提取图像的边缘特征，cell 大小与 block 大小选择为 NxN 方形单元，例如 8x8、16x16、32x32，经过算法实验验证，本章选择 cell 大小为 8x8 时，block 大小为 32x32 综合性能最优越。

④HOG 边缘特征作为 SVM 分类器的特征向量。

⑤SVM 分类器对训练集图像进行训练并保存，选择的 SVM 为软间隔 SVM，惩罚系数选择为 1。

⑥使用测试集图像数据对训练好的分类模型进行测试，评估模型的分类识别效果。

⑦将分类模型对测试集图像的分类结果统计在混淆矩阵中。

最后得到的混淆矩阵如下：

表 2-2 分类结果

Confusion Matrix		分类值	
		Class 0	Class 1
真实值	Class 0	9	1
	Class 1	4	6

根据表 4.3 数据，可以计算模型的准确率为：

$$accuracy(class0) = \frac{9}{1+9} = 90\%$$

$$accuracy(class1) = \frac{6}{4+6} = 60\%$$

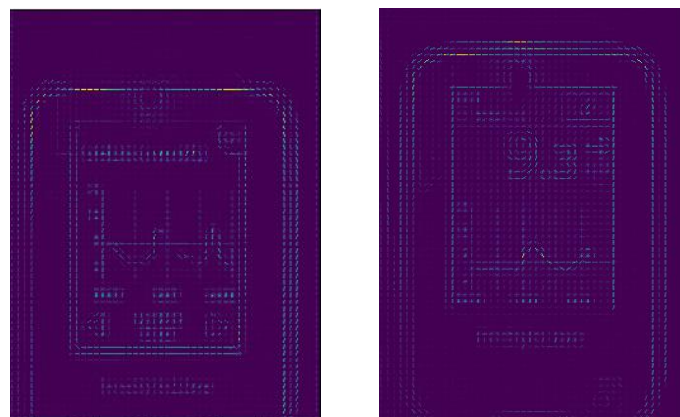
$$accuracy = \frac{9+6}{1+9+4+6} = 75\%$$

虽然整体模型准确率可以达到 75%，但是对于 8 小时图像的准确率比较低，只有 50%，因此整个模型还有较大改进空间，因此还需要进一步修改模型。

2.5 改进 HOG 特征的 SVM 分类实验

从上面的实验结果可以发现，单纯的使用 HOG 特征进行识别效果并不是很理

想。HOG 特征主要提取的是边缘特征，而多种 CGM 图像的边缘特征比较近似，仅通过边缘特征区分两类图片有一定难度。因此所以需要考虑对 HOG 特征加以改进。



a)24 小时图像 HOG 特征 b)8 小时图像 HOG 特征

图 2-9 不同类的 HOG 特征对比

通过观察我们可以发现，多种 CGM 图都由边框与屏幕组成。边框基本相近，主要是屏幕部分图像不同，所以设计的特征需要能够提取屏幕部分图像的信息。这里设计的一种能够体现图像灰度分布的特征描述子。

对于改进的特征来说，我们希望它能够体现出图像的灰度分布，因为同一类图像的显示内容接近，因此灰度分布也比较接近，同时需要对光照与位置不敏感，使得在不同光照与拍摄角度下同一类图像的特征也比较接近。这里设计的特征计算包括预处理与归一化两步：

(1) 预处理：预处理先要灰度化，并将图像缩小到一个较小的尺寸，通过这种方式减少图像的非显著特征与干扰。灰度化之后可以对图像进行形态学腐蚀运算。腐蚀运算的主要目的是想减少强光照部分，在光照尤其强光干扰下会使改变图像的灰度分布。下一步对图像均值滤波操作使图像模糊，并且滤波的卷积核要比较大，这里选择为 9×9 。通过均值滤波，可以在图像去噪的同时也破坏了图像的细节部分，这样得到的结果就可以体现图像的灰度分布。

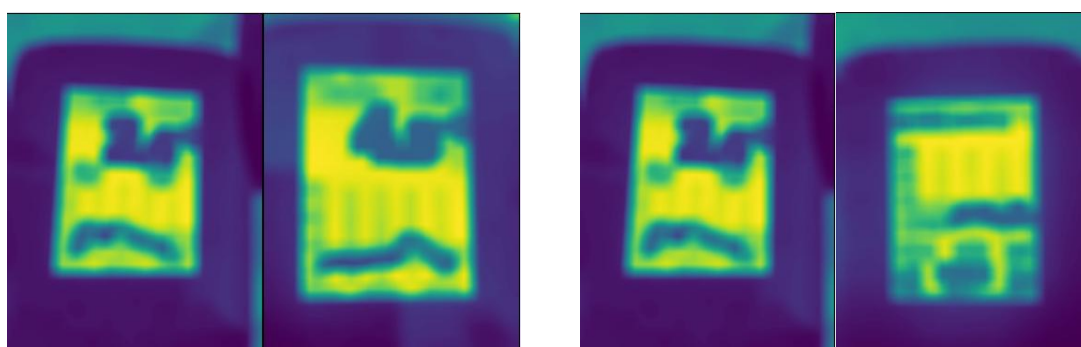


图 2-10 灰度分布特征预处理

(2) 归一化与灰度变换：归一化的目的主要是进一步减小光照影响。首先将灰度像素值线性变换到 0-1。之后将图像灰度通过 Sigmoid 函数进行计算。Sigmoid 函数表达式为 $f(x) = \frac{1}{1+e^{-x}}$ ，Sigmoid 函数的输出仍在 0-1 之间，但是小于 0 部分更接近 0，大于 0 部分更接近 1。这里对 Sigmoid 函数进行修改，图像灰度需要先减整幅图像的灰度均值，再乘上系数 3。

$$I(x, y) \leftarrow \frac{1}{1+e^{-3 \times (I(x, y) - \text{average})}} \quad (2-12)$$

最后再次将图像的灰度归一化，使得得到的结果均值为 0，标准差为 1。



a)同类间灰度分布特征对比

b)不同类间灰度分布特征对比

图 2-11 灰度分布特征结果

通过对得到的灰度分布特征结果看到，同一类的特征比较接近，不同类的特征有比较大的差别。将灰度分布特征展开为一列的向量，因此原来 75×100 的特征数据转化 7500 维向量，并于计算得到的 HOG 特征向量拼接到一起，生成最终的图像特征向量。

对于改进 HOG 特征使用 SVM 分类算法进行分类，首先训练 SVM 分类器，再使用测试集对模型进行测试，得到最终的混淆矩阵对算法进行评估。改进后的分类模型的工作流程如下图。

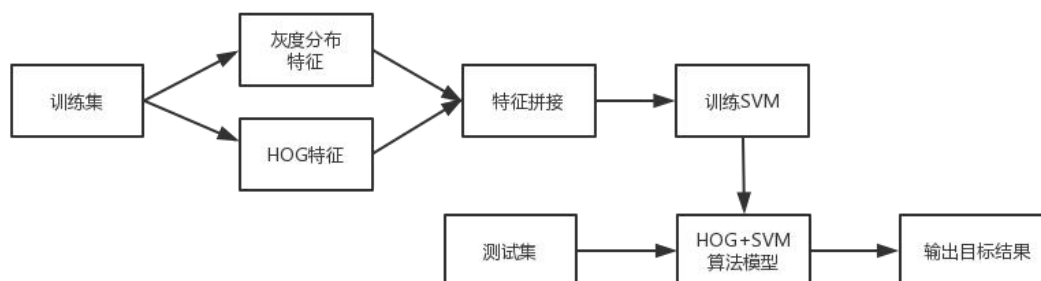


图 2-12 改进 HOG 特征分类模型工作流程

通过该流程对模型进行评估，最终测试得到的混淆矩阵结果如下：

表 2-3 分类结果

Confusion Matrix		分类值	
		Class 0	Class 1
真实值	Class 0	9	1
	Class 1	1	9

改进后的模型的最终分类准确率达到了 90%，由此可以看到，通过改进 HOG 特征进行支持向量机分类，提取的特征包含将边缘信息与灰度分布信息，与原模型相比明显提高了分类效果。

2.6 本章小结

本章主要进行在小样本场景下的 CGM 血糖仪屏幕图像分类任务研究。首先使用基于 HOG 特征的 SVM 分类算法，根据 HOG 特征的原理，提出基于改进 HOG 特征的 SVM 分类算法，并最终在数据集上达到 90%，相较于原始 HOG 特征有较大提升。

第三章 图像校正

3.1 引言

本章任务主要是对上一章以及分类识别后的 CGM 血糖仪屏幕图像进行校正。校正主要包括分割屏幕图像与投影变换校正两个部分。得到校正后的屏幕图像，可以更有效地提取数据。对于上一章的两类 CGM 屏幕图像，校正算法基本一致，接下来以上一章的 24 小时图像为例说明校正算法。

3.2 分割屏幕图像

3.2.1 区域面积重采样插值

在分割屏幕图像之前的第一步是对图像的预处理。这里预处理主要是对图像进行缩小。一般手机拍照的图像比较大，一般高度大于 4000，宽度大于 3000，且比例为 4:3。为了提高运算速度，需要先将图片缩小到一个较小的尺寸，比如 600x800。缩小图像一般是通过插值实现的，使用比较多的方法有最邻近法、双线性内插法、三次内插法等。但是使用这些方法在图像缩小系数比较大的时候会产生明显的干扰波纹，严重影响后面的检测。因此这里采用的插值方法是区域重采样插值，原理是使用像素面积关系重采样。

所谓像素面积重采样，指的是找出像素点对应的原来部分求有由像素面积计算加权平均值，具体实现分为两种情况：

(1)缩放是整数倍：那么直接计算对应部分的像素值的均值即可。例如下图，缩放倍数为 2，那么缩小后的点的像素值是红色区域的像素均值。

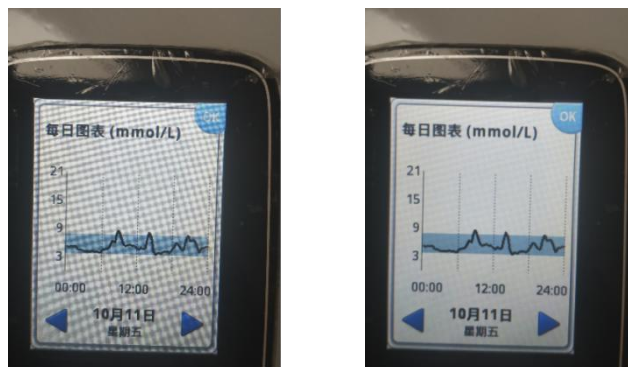
1	2	3	4	1	2	3	4
2	3	4	5	2	3	4	5
3	4	5	6	3	4	5	6
4	5	6	7	4	5	6	7

图 3-1 像素面积重采样示意图

(2)缩放不是整数倍：那么需要进行根据像素面积进行加权计算。例如下图中，缩放倍数为三分之二，那么缩放后红圈对应的像素值计算方式为

$$I(0,0) = \frac{1 \times 1 + 2 \times 0.5 + 2 \times 0.5 + 3 \times 0.25}{1.5 \times 1.5} \quad (3-1)$$

可以看到通过区域面积重采样插值，可以明显减少图像缩小后的干扰波纹，便于后续的图像分割与视觉检测。



a)其他插值方法效果 b)区域重采样插值效果

图 3-2 不同插值方法缩小效果

3.2.2 GrabCut 分割原理

Grabcut 是一种图论的分割方法，起源于 Graphcut 方法。Graphcut 方法把图像分割问题进行转化，将原问题转化为图的最小割问题。

首先待分割的图像用无向图 $G=\langle V, E \rangle$ 表示， V 和 E 为图的全部顶点与边组成的集合。Graphcut 的图将全部的像素点作为图的普通顶点，并在此基础上增加了多了两个终端顶点，分别为 S 与 T 。所有普通顶点都要与这两个终端顶点相连。因此 Graphcut 包含两类顶点与两类边。

第一类：第一类顶点也就是普通顶点，为图像中的每个像素点组成。每两个相邻顶点，也就是图像中每两个邻域像素的连接属于第一类边，称为 n -links。

第二类：两个终端顶点也就是 S 和 T 为第二类顶点。 S 和 T 两个终端顶点与全部普通顶点相连，连接的边为第二类边，称为 t -links。

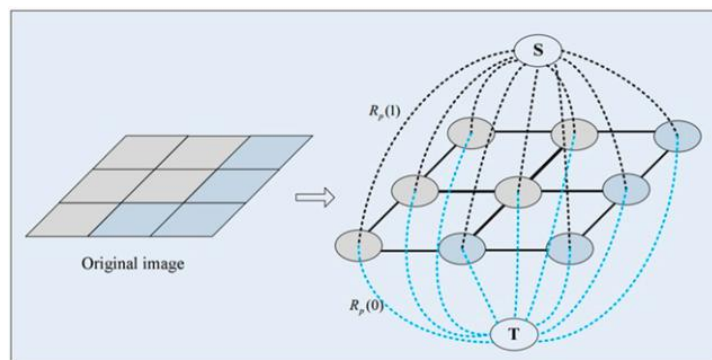


图 3-3 Graphcut 切割图模型

在 Graphcut 中, Cuts 代表某种边的集合, 该集合内元素为以上定义的两边, 同时满足该集合中所有边分割后, S 和 T 图两部分不相连, 因此被定义割, 代表整个图被分割为两部分。如果一个割满足它的边的所有权值之和为全部的割最小, 那么这个割就被定义为最小割, 图割的计算目的也就是获得最小割。这个最小割将全部图的顶点归类 S 和 T 两个对立的集合, 满足 $S \cap T = \emptyset$ 且 $S \cup T = V$ 。最终 S 与 T 两个子集分别为图像的前景部分集合和背景部分集合, 这样就实现了图像的分割。假设最后的分割结果为 L, 则待优化的能量为:

$$E(L) = \alpha R(L) + B(L) \quad (3-2)$$

其中, $R(L)$ 代表区域项, $B(L)$ 代表边界项, 而 α 为超参数, 代表区域项和边界项两者对能量大小影响的比重。如果 α 为 0, 那么能量就只有边界项, 没有区域项。Graphcut 就是通过优化能量, 使能量函数达到最小值, 进而得到图像的分割。对于 $R(L)$ 的计算, Graphcut 使用灰度直方图模型对背景建模, 因此是通过比较像素的灰度和给定的前景的灰度直方图来计算。 $B(L)$ 主要代表分割 L 的边界惩罚, 因为图像分割边界两侧应该差别比较大, 所以分割边界两侧的两点 p 和 q 越相似, 那么应该 $B(L)$ 越大, 如果它们非常不同, 那么 $B(L)$ 就接近于 0^[33]。判断相似与否可以通过灰度差计算。

Grabcut 在 Graphcut 的基础上进行了一系列修改, 实现了分割效果提高。具体改进包括: 目标背景上 Grabcut 通过在 RGB 三通道建立高斯混合模型进行背景建模, 改进了 Graphcut 的背景建模也就是灰度直方图模型; Graphcut 直接一次计算实现能量最小化, 而 Grabcut 计算过程为迭代过程, 需要不断计算分割估计, 并且同时不断学习模型的参数, 也就是高斯混合模型的参数; Graphcut 需要预先确定种子点为前景活北京, 但是 Grabcut 只需要指定背景区域的部分。Grabcut 的能量函数为:

$$E = U + V \quad (3-3)$$

其中, U 代表区域项, 在通过高斯混合模型进行背景建模之后, 可以求出某个点属于前景的概率, 而 U 为概率的负对数, 表示通过背景模型归类得到的结果的惩罚。混合高斯密度数学模型如下:

$$D(x) = \sum_{i=1}^k \pi_i g_i(x; \mu_i, \Sigma_i) \quad (3-4)$$

其中:

$$\sum_{i=1}^k \pi_i = 1 \quad 0 \leq \pi_i \leq 1 \quad (3-5)$$

$$g(x; \mu, \Sigma) = \frac{1}{\sqrt{2\pi^d |\Sigma|}} e^{-\frac{1}{2}(x-\mu)^T \Sigma^{-1} (x-\mu)} \quad (3-6)$$

因此:

$$D_i = -\log \pi(\alpha_i, k_i) + \frac{1}{2} \log(|\Sigma(\alpha_i, k_i)|) + \frac{1}{2} (x_i - \mu(\alpha_i, k_i))^T \Sigma^{-1} (x_i - \mu(\alpha_i, k_i)) \quad (3-7)$$

$$U = \sum_{i=1}^k D_i \quad (3-8)$$

而边界能量项 V 体现邻域像素 m 和 n 之间不连续的惩罚，其公式为:

$$V = \gamma \sum_{m,n} [\alpha_n \neq \alpha_m] e^{-\beta \|z_m - z_n\|^2} \quad (3-9)$$

一般而言，参数 β 取决于图像的对比度，常数 γ 根据经验一般为 50。算法运行时需要人工指定前景预选框，即图像框外的部分指定为图像背景，而框内的存在着图像需要分割出的前景部分。运行初始化时需要使用 k-means 聚类初始化，之后就可以迭代计算图的最小割，从而获得图像的分割^[33]。

3.2.3 改进 GrabCut 分割

GrabCut 分割算法广泛应用于抠图，但是在 GrabCut 算法无法实现自动端到端的分割。这是因为该算法需要人工选择预选框，并且可以看到该方法的分割效果与初始预选框密切相关。



图 3-4 预选框选取失败效果

为了系统自动分割屏幕图像，需要自动计算预选框，这里提出一种用于区域生长的计算预选框的改进的 Grabcut 算法，适用于分割这种具有明显边框的图像。改进 GrabCut 分割包括以下步骤:

(1)获取图像梯度：为了避免光照的影响，选取的是梯度信息进行处理。因此先通过 Sobel 算子计算梯度，包括水平方向与垂直方向两个方向，计算每个点的梯度幅值，最后获得图像的梯度图像。

(2)Otsu 阈值分割：Otsu 分割算法通过计算最大类间方差实现分割。任意选取一个灰度值 T ，则可以将这个直方图分成前后两部分。设图像的灰度平均值为 u ，前景部分在图像中比例为 w_0 ，灰度均值为 u_0 ；背景部分在图像中比例为 w_1 ，灰度均值为 u_1 。那么类间方差为：

$$ICV = w_0(u - u_0)^2 + w_1(u - u_1)^2 \quad (3-10)$$

可以通过遍历 T 的各种取值计算最大类间方差，获得的阈值也就是图像分割的阈值。

(3)形态学膨胀：主要目的是为了防止边界框不闭合。如果屏幕图像的边缘部分不闭合的话，那么无法用来进行区域生长的计算。

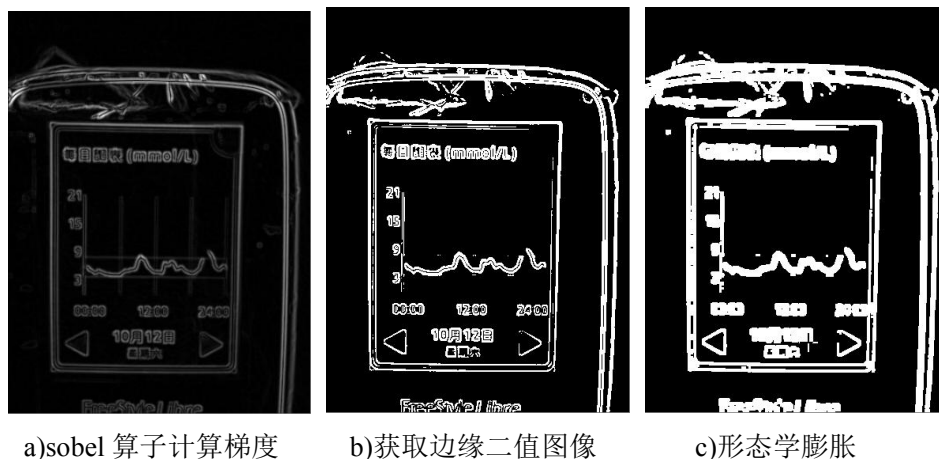


图 3-5 检测屏幕边缘

(4)区域生长获取预选框：区域生长算法的基本思想是合并有相同属性的像素点。区域生长开始计算前需要指定种子点作为区域生长的起点，比较种子点与邻域内点是否具备相同属性，将邻域内属性相同的点进行合并实现区域的向外生长，直到图像内没有可以继续包含在区域内进行生长，这样完成全部的区域生长的计算^[34]。计算时有两个问题需要被确定：

(a)种子点的选取：种子点的选取很多时候都采用人工交互的方法实现，一般是要选择到物体内部。为了能够实现自动检测，可以通过先验信息去选择，如物体在图像中的位置，图像的像素值分布。这里选择图像的中心附近作为种子点，这是因为拍摄的屏幕一般能够包含图像中心。

(b)在生长过程中包括相邻像素的标准：由于是通过二值图像进行生长，这里将

附近为 0 像素值的点选取进来。

(5)对区域生长结果的处理：处理主要包括两步，一是需要将得到的分割结果的空洞填补上，如果存在这些空洞将会影响后续的 GrabCut 分割；二是要将填补后的分割结果进一步形态学膨胀，这一步主要考虑的是这里区域生长是由内向外的生长，最后的分割结果可能会无法覆盖整个屏幕部分图像，所以需要对分割结果进行形态学膨胀，以确保最终得到的预选框能够覆盖整个屏幕。

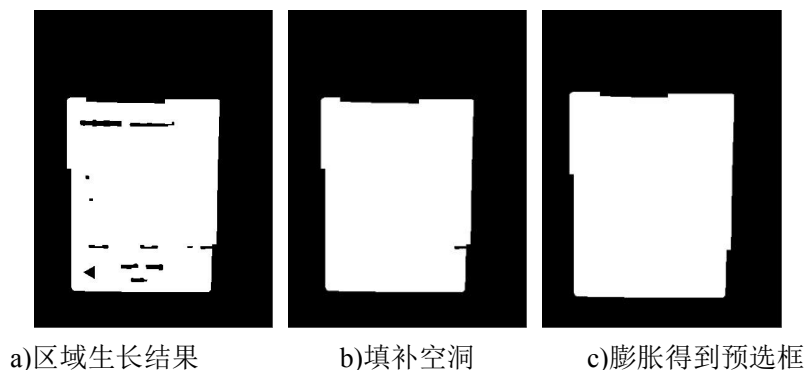


图 3-6 区域生长设置预选框

(6)GrabCut 分割：在得到了最后的预选框之后得，就可以将预选框指定为图像可能的前景，作为 GrabCut 算法的初始条件进行计算，就可以得到屏幕图像分割结果。

完整的屏幕分割算法流程图如下：

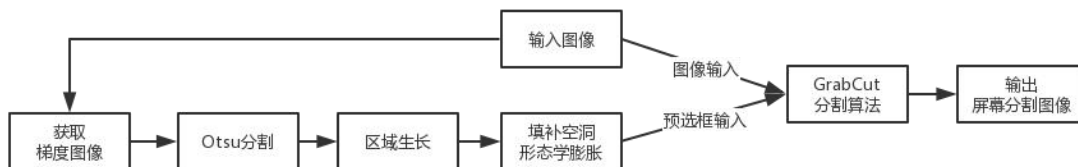


图 3-7 屏幕分割算法

相比于原始的 GrabCut 算法，这里通过区域生长进行设置预选框的改进方法，使原算法更适用于视觉检测，同时比人工设置预选框更加精确，能够提高分割的结果，同时也能提高接下来提取数据的效果。

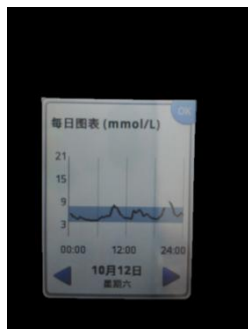


图 3-8 分割屏幕结果

3.3 投影变换校正

通过上一步分割，得到了整个屏幕部分的分割，但在提取数据之前，还要进行进一步校正，因为我们分割出来的屏幕轮廓并不是矩形而是一个凸四边形。这是因为拍摄实际上是一个投影的过程，拍摄时无法做到拍摄的手机平面与 CGM 血糖仪屏幕保持完全平行，导致拍摄出的屏幕会产生形变，屏幕的轮廓不再保持矩形。考虑到后面提取数据效果，需要进行投影变换进行校正。

3.3.1 投影变换原理

通常从三维空间物体转变到二维的平面图像过程中发生了投影变换。人们的日常生活中处处存在着投影现象。例如，日常生活中我们使用手机中的拍照功能，是将实际的三维立体实物投影到相机成像平面转变成二维平面图像。

投影变换是将图片投影到一个新的视平面^[35]，具体来说就是利用投影中心、原平面上的点、变换后的点三点共线的条件，将透视面绕透视轴旋转某一角度后，对应点坐标的变换关系。

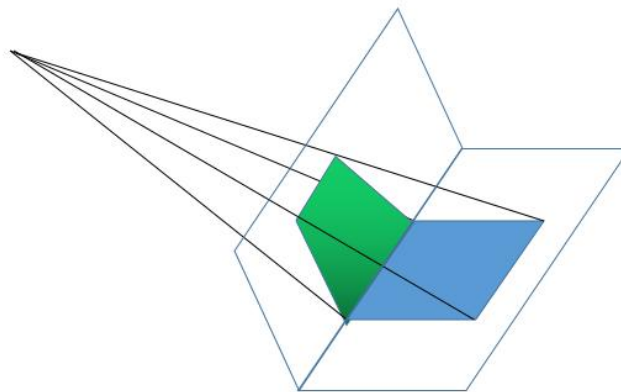


图 3-9 投影成像

根据投影中心和投影面，可以分为两类：一类是平行投影，投影中心到投影面的距离是无限的，投影线互相平行；另一类为透视投影：投影中心到投影面的距离是有限的，投影线不平行。投影变换其主要特点是，直线经变换后仍然是直线，但平行的直线经变换后却可能相交。例如两条铁轨，在三维空间为平行的两条直线，但是人眼或相机拍摄看到的就两条直线最终交汇到一点。

在计算投影变换公式之前需要先进行定义点的齐次表示。一般来说，平面上一个点用二维坐标表示，用矩阵表示一个线性变换，则整个变换表示矩阵乘法为：

$$\begin{bmatrix} x \\ y \end{bmatrix} = A \begin{bmatrix} u \\ v \end{bmatrix} \quad (3-11)$$

但是这种方法有个问题，就是无法用矩阵乘法表示带有平移的变换。这是因为矩阵乘法表示的线性变换要求原点变换后还是原点。为了将带有平移的各种变换表示为矩阵乘法，我们可以用齐次坐标表示一个点，将坐标扩展一个维度，点表示为 (x, y, w) ，该点代表笛卡尔坐标系中的 $(\frac{x}{w}, \frac{y}{w})$ ，其中 w 为0时，代表该点位于无穷远处。例如 $(2, 4, 1)$ 与 $(4, 8, 2)$ 一样，都代表 $(2, 4)$ 这个点。通过这种方式就可以用矩阵乘法代表多种变换，例如将原点平移到 $(1, 1)$ 的变换，可以用齐次坐标与矩阵乘法表示成：

$$\begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (3-12)$$

根据射影几何中的定理，可以得到投影变换在其次坐标下为：

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad (3-13)$$

将矩阵展开，可以得到变换前后的坐标关系为：

$$u = \frac{a_{11}x + a_{12}y + a_{13}}{a_{31}x + a_{32}y + a_{33}} \quad (3-14)$$

$$v = \frac{a_{21}x + a_{22}y + a_{23}}{a_{31}x + a_{32}y + a_{33}} \quad (3-15)$$

根据射影几何中的定理，投影变换前后对应的四对不共线像素点坐标，即可求得透视变换矩阵；反之，给定透视变换矩阵，即可对图像或像素点坐标完成透视变换。因此我们需要确定变换前后四对像素点坐标，计算出投影变换的矩阵；再通过投影变换矩阵，计算出变换后的图像像素。

3.3.2 轮廓拟合

计算投影变换矩阵需要前后四对像素点，这里我们指定为变换前的四个点是屏幕边框的四个顶点，变换后的四个点是图像的四个角落。计算投影矩阵还需要指导四个点的变换前后坐标，因此在校正之前还需要确定屏幕的四个顶点坐标。屏幕的四个顶点为屏幕边缘轮廓的四个顶点，考虑到在投影后矩形轮廓不再为矩形而是一个凸四边形，因此需要将屏幕轮廓拟合为一个四边形，这样屏幕的四个顶

点坐标也就是拟合出来的四边形的四个顶点坐标。

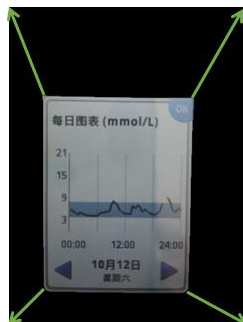


图 3-10 投影的对应点

轮廓拟合包括以下步骤

(1) 图像转换为二值图像：由于 GrabCut 分割得到的图像将背景设为 0，因此这里选择最简单的阈值二值化即可。

(2) 通过二值图像获取轮廓：对于二值图像获取轮廓比较简单，具体来说就是遍历图像每个像素点，查看附近连通域内 9 个像素的最大值与最小值。如果最大值与最小值不同，那么证明该点位于边缘上，这样就去欸的那个了轮廓图像。

(3) 轮廓进行四边形拟合：拟合方法选择的是 Douglas-Peucker 算法，该算法可以将轮廓用一些点去近似表示。该算法最开始提出是将曲线进行简化成一系列点，基本步骤如下：

- a) 连接曲线两个端点 A 与 B，获得直线 AB；
- b) 获取曲线上离直线 AB 最远的点 C，同时求出 C 与直线 AB 的距离；
- c) 若给定的阈值大于 C 与直线 AB 的距离，则以该直线作为曲线的近似，曲线拟合计算结束；
- d) 若不满足 c 中的条件，则曲线分为 AC 和 BC 两段，并对两段曲线分别进行 a-c 的计算；
- e) 曲线计算结束后，连接每段曲线的首尾形成折线，即原曲线的近似结果^[36]；

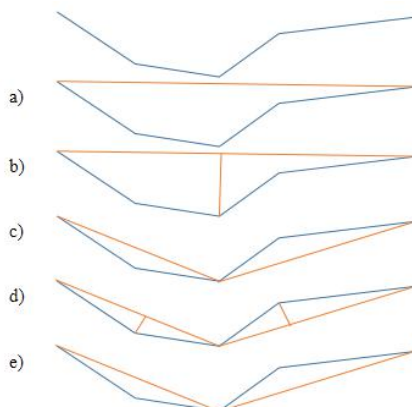


图 3-11 DP 算法示意图

这里使用 DP 算法时需要进行两点修改。第一，与简单拟合曲线不同，这里要拟合的是闭合曲线，因此第一步的初始化需要进行修改。原始算法初始化的两个点为直线的两端，这里修改为变成闭合轮廓曲线上最远的两个点，这样就可以将该算法应用到多边形拟合。第二，目标是想将轮廓曲线拟合成四边形，但是 DP 算法输入参数是阈值，也就是拟合后的曲线与原曲线最大的距离，因此是通过逐渐增大输入参数不断尝试，直到最后拟合出的曲线是四边形。

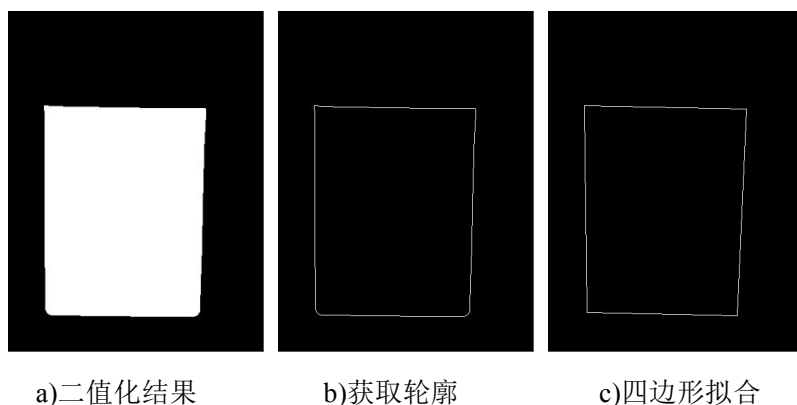


图 3-12 四边形拟合校正过程

完整的轮廓拟合算法流程图如下：

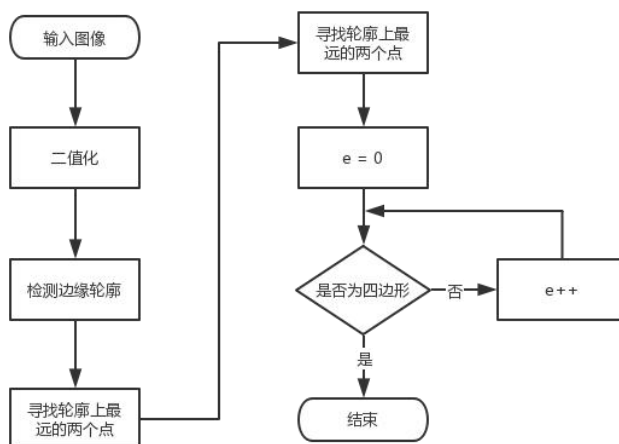


图 3-13 轮廓拟合算法流程图

3.3.3 投影变换校正

在拟合结束后，就可以找到屏幕的四个顶点，就可以计算对应的的投影变换矩阵了。如果我们想从变换前后的对应点去求这个投影变换，那就需要解线性方程组解出这公式(3-14)与(3-15)中的 9 个系数。

但是转化后的线性方程组是齐次方程组，方程组的系数矩阵的秩在四点不共线的情况下为 8，未知数个数大于系数矩阵的秩，因此方程有无穷解。也就是说两组

不同 9 个系数可能确定的是同一组变换，例如变换矩阵全体系数乘 2，表达的变换仍然不变。所以这里设置 $a_{33} = 1$ ，这样要求解八个未知数，对应的系数矩阵的秩为 8，这样线性方程组恰好有唯一解。

有了这个方法，如果我们想通过投影变换使 a, b, c, d 变换到 m, n, p, q 四个点，我们可以得到通过该方法计算出 m, n, p, q 到 a, b, c, d 这个投影变换的矩阵。所以我们可以的获得图像四个顶点到轮廓四边形四个顶点的投影变换矩阵，通过投影变换矩阵，我们可以得到变换前后的像素值对应关系为：

$$I(x, y) = I_0(u, v) \quad (3-16)$$

其中 u, v 与 x, y 的变换关系为公式(3-14)与(3-15)。但是 (x, y) 点对应的 (u, v) 可能并不是整数，因此最后还需要考虑插值的问题，这里选择比较简单的双线性插值。

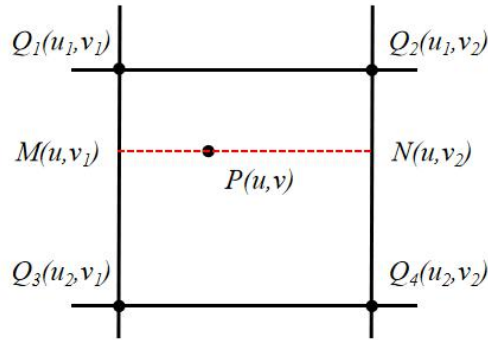


图 3-14 双线性插值

假设计算出的对应点为 P ， P 最相近的四个像素点分别为 Q_1 ， Q_2 ， Q_3 与 Q_4 。过 P 向水平方向作垂线，得到 M ， N 两个点，先用 Q_1 与 Q_3 插值得到 M 点的值，再通过 Q_2 与 Q_4 插值得到 N 点的值，最后使用 M ， N 两个点对 P 插值得到 P 点像素值，计算公式为：

$$\begin{aligned} f(P) = & \frac{(u-u_1)(v-v_1)}{(u_2-u_1)(v_2-v_1)} f(Q_1) + \frac{(u-u_1)(v_2-v)}{(u_2-u_1)(v_2-v_1)} f(Q_2) \\ & + \frac{(u_2-u)(v-v_1)}{(u_2-u_1)(v_2-v_1)} f(Q_3) + \frac{(u_2-u)(v_2-v)}{(u_2-u_1)(v_2-v_1)} f(Q_4) \end{aligned} \quad (3-17)$$

计算出的像素值也就是变换后 (x, y) 点对应的像素值，这样就通过投影变换计算出变换后图像。

3.4 校正算法实验验证

最终校正得到的图像结果如下：

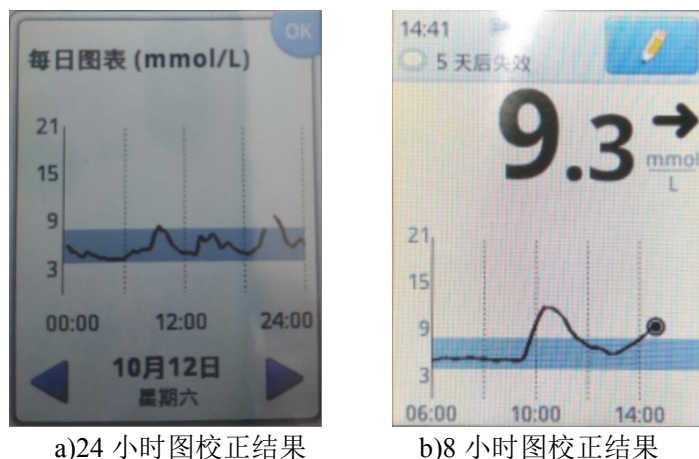


图 3-15 校正结果

按照本章的算法设计，将校正算法中各部分整合到一起，形成完整的图像校正算法。本小节通过对数据集图像进行测试，对最终的校正算法进行了验证，并根据准确度、运行时间两个评价指标对其进行评估。

(1) 准确度：

通过对 40 张测试图像进行测试，最后有 38 张图像被成功校正，2 张无法通过校正算法进行校正，成功率有 95%。

使用无法校正的图像对算法每一步进行测试，发现失败的原因在分割屏幕失败，由于不均匀强光照，无法获取的有效的边缘，在算法第一步进行边缘检测时，导致区域生长计算预选框失败，进而导致屏幕图像分割失败。因此在整个校正算法在边缘检测部分还有提高余地，可以使用其他改进的边缘检测算法进行该步的边缘检测，进一步提高成功率。

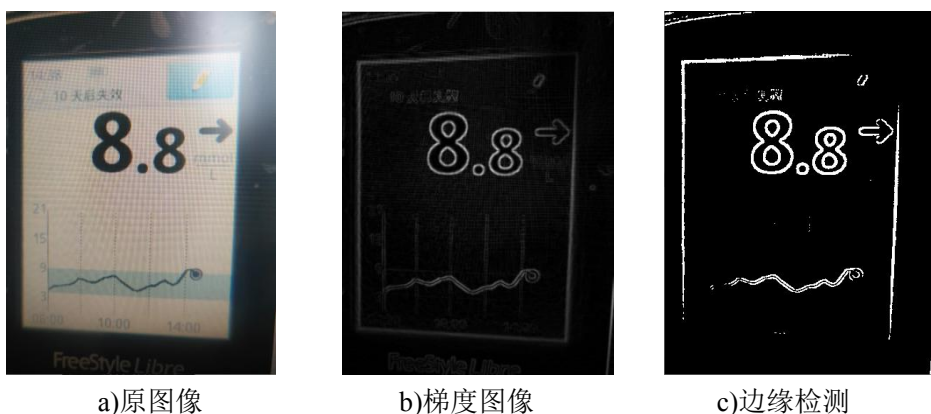


图 3-14 预选框计算失败

(2) 运行时间：

记录每张图像校正的时间，最后算出平均每张图像计算时间为 1.144 秒。其中运行最长时间为 1.602 秒，运行最短时间为 0.725 秒，由于并非为实时检测条件，

因此运行速度可以达到实际使用的要求。

3.5 本章小结

本章任务内容包括屏幕图像分割与图像校正两大部分。屏幕图像分割主要使用改进的 GrabCut 算法进行分割，通过边缘检测、区域生长计算出前景预选框，将预选框放入 GrabCut 算法中，最终实现自动的图像分割，并且该算法通过验证得到的成功率为 95%。图像校正主要是通过投影变换实现，通过计算投影变换矩阵，找出变换前后的坐标变换关系，就可计算出变换后的图像。实际计算时还要考虑变换对应点不是整数点，那么需要通过双线性插值计算出最后的图像像素值。

第四章 图像数据视觉检测

4.1 引言

在得到上一章的校正后的图像后,就可以进行数据的视觉检测。获取包括两种数据,一是血糖曲线数据,另一种是时间与日期数据等。血糖曲线数据主要靠检测关键点进行校正,分割曲线图像,计算得到最后的血糖值。时间与日期数据等作为文字信息,使用目前常用的深度学习算法进行获取场景文本识别,包括文本检测与文字识别两个步骤。

4.2 血糖数据视觉检测

血糖曲线视觉检测主要是通过校正后的图像,检测出实时血糖数据,辅助医生查看出具体血糖数据进行远程医疗。

4.2.1 检测坐标系与血糖曲线分割

这里检测血糖数据主要通过传统图像处理方法检测对应的关键点实现。关键点的设置依赖于先验信息,也就是人工挑选待检测的特征。

在识别分割曲线数据之前,需要进行坐标系的识别,只有检测到坐标系位置,才能计算血糖曲线的相对位置,将后续分割到的数据进行读数,获取具体的数值。这里就是选取特征点为左侧坐标轴的两个端点与最右侧虚线的两个端点。

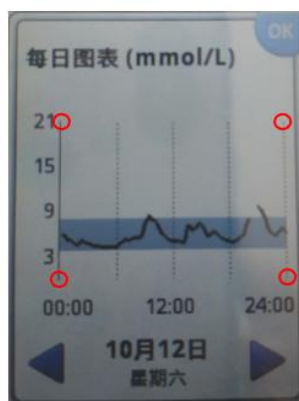


图 4-1 坐标系特征点

左侧的两个特征点位于坐标直线两端,因此主要思路是通过 Hough 变换检测出坐标轴位置,再用灰度去阈值分割,找出两个端点。具体检测步骤包括以下流程:

(1)获取左半平面水平梯度图像：这里主要是想检测纵坐标，因此只用 Sobel 算子检测水平方向梯度，检测之后用 Otsu 法分割出边缘。因为坐标轴一定在左半平面，所以可以将右半平面和最左侧边缘部分置零，去掉无用的部分。

(2)初步检测坐标轴位置：使用 Hough 变换检测直线。但是 Hough 变换无法检测到直线的端点，并且 Hough 变换输入的参数为判断为直线的域值，所以这里不断尝试减小输入参数，直到无法检测出直线。对得到直线图像与原有的边缘图像取与运算，这样就可以检测到坐标轴可能存在的大体位置。但是还会存在写其他干扰，因为坐标是比较长的部分，所以去掉上下与中间不相连的区域，这样就可以初步筛选到坐标位置。

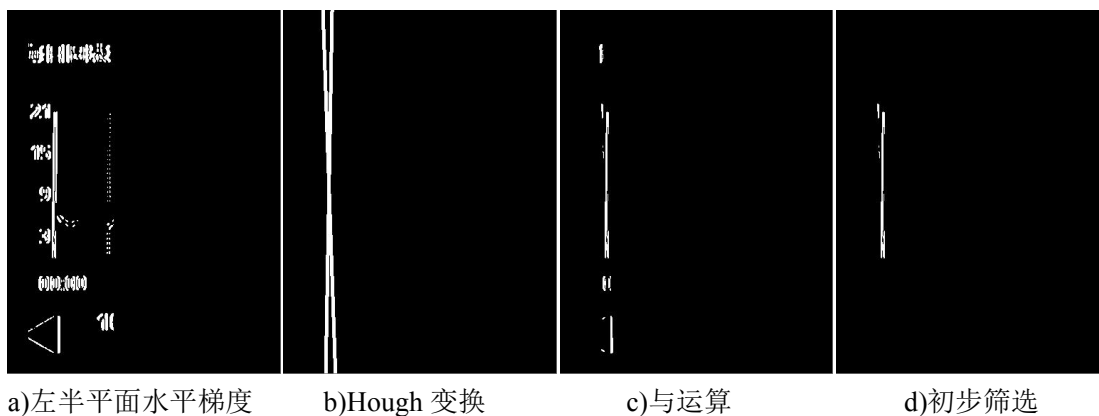


图 4-2 初步筛选坐标轴

(3)阈值分割：将初步筛选后的结果转为灰度图片，之后将前景与背景分割开，前景主要为坐标轴与坐标轴左侧的数字。考虑到这部分灰度值比背景差距表达，可以直接用灰度值进行分割。为了保证用灰度值进行分割的效果，可以先进行灰度变换，这里使用的线性变换，将灰度归一化，这样调整对比度就优化阈值分割的效果。

$$f(x) = \frac{255 \times (x - \min)}{\max - \min} \quad (4-1)$$

(4)准确筛选坐标：在得到分割出的数字与纵坐标轴之后，再次用 Hough 变换检测坐标轴，不断调整阈值，最后就得到了纵坐标轴的准确位置。Hough 变换得到的结果与之前阈值分割的结果进行与运算，就可以完全分割出纵坐标。左侧两个特征点取刚才分割出的纵坐标最上面和最下面的两个点。

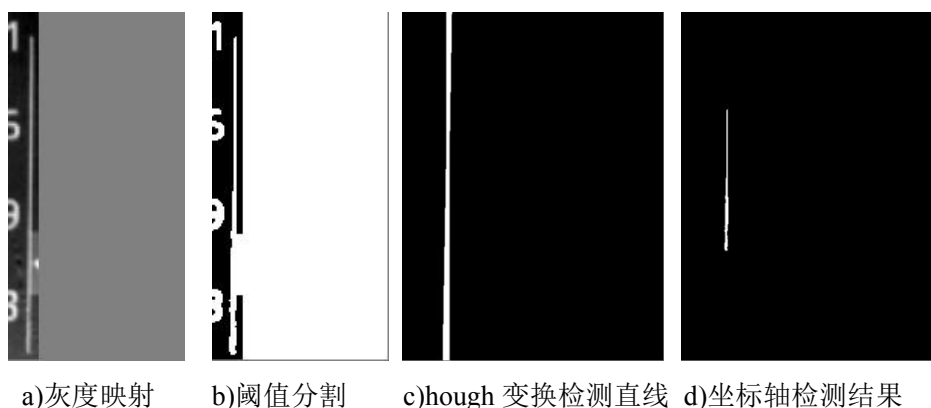


图 4-3 检测坐标轴及端点

在检测右侧两个特征点之前，可以检测蓝色区域范围辅助检测，这是因为右侧特征的一定不会超过蓝色区域太多。因为蓝色区域高度一定在坐标轴之内，可以现根据上一步得到的左侧坐标点截取出这部分。考虑到蓝色区域 B 分量比 R 分量，可以用截取出来部分的 B 通道减去 R 通道得到通道差值，然后计算梯度就可以得到上下边缘。对于检测出的边缘图像需要再进行形态学开运算，去掉一些孤立噪点，就可以得到完成的蓝色区域上下边缘，也就获得了蓝色区域最右侧边缘的坐标。



图 4-4 检测蓝色区域

右侧特征点位于屏幕右半平面，并且在蓝色区域最右侧附近，同时高度与左侧特征点接近，因此可以先裁剪出这部分。因为这些部分都是虚线，这里使用 Harris 角点检测检测虚线位置。右上角的特征点位于最右侧虚线的上端，所以先检测最右侧的虚线，截取对应区域。找到之后可以再次用灰度线性变换加强对比度，再用灰度值进行分割这样就检测到右侧的虚线，右侧的特征点位于虚线的最上端。右下角虚线的端点检测方法相同，不再赘述。



图 4-5 检测虚线

完整的检测坐标轴关键点完整步骤如下：

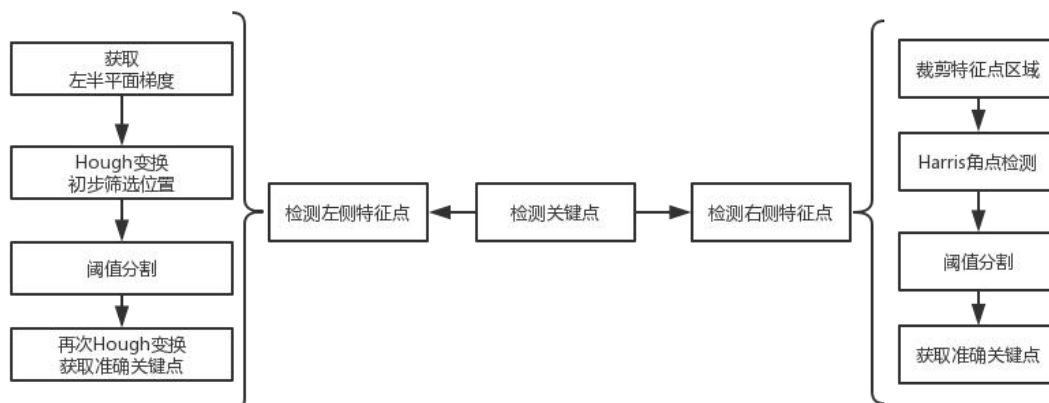
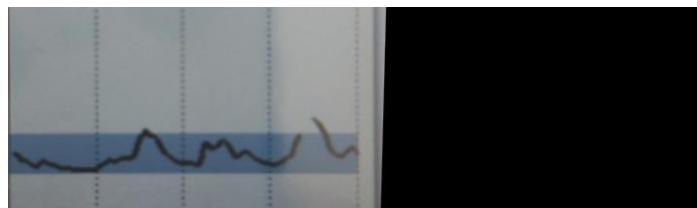
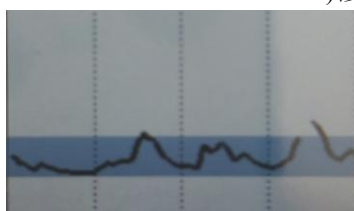


图 4-6 坐标轴关键点检测

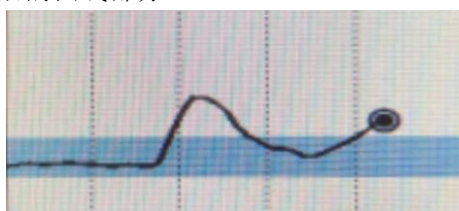
在检测到四个特征点之后，由于第一步分割存在误差，检测到的坐标并不是竖直方向，而是有一定误差，因此需要考虑进一步校正。为了减小误差再次用投影变换再次校正，将四个特征点投影到四个角落，这样就可以得到修正后的曲线。裁剪蓝色区域以外的部分，这样就得到标准的血糖曲线，并且右下角与坐标轴零点对齐，水平与竖直方向也经过了校正。



a)修正后的曲线部分



b)24 小时曲线图像



c)8 小时曲线图像

图 4-7 获取曲线图像

将血糖曲线图像预处理，转为灰度图像并反转进行。可以看到曲线部分的灰度值比较大，但由于有不均匀光照导致不同位置的曲线灰度相差比较大。为了减少光照影响，这里将每一列减去对应列的蓝色区域的值，这样就除了曲线部分，剩下的灰度值比较小，可以用灰度值进行阈值分割，分割之前可以用灰度线性变换进一步加强对比度，增强分割的效果。

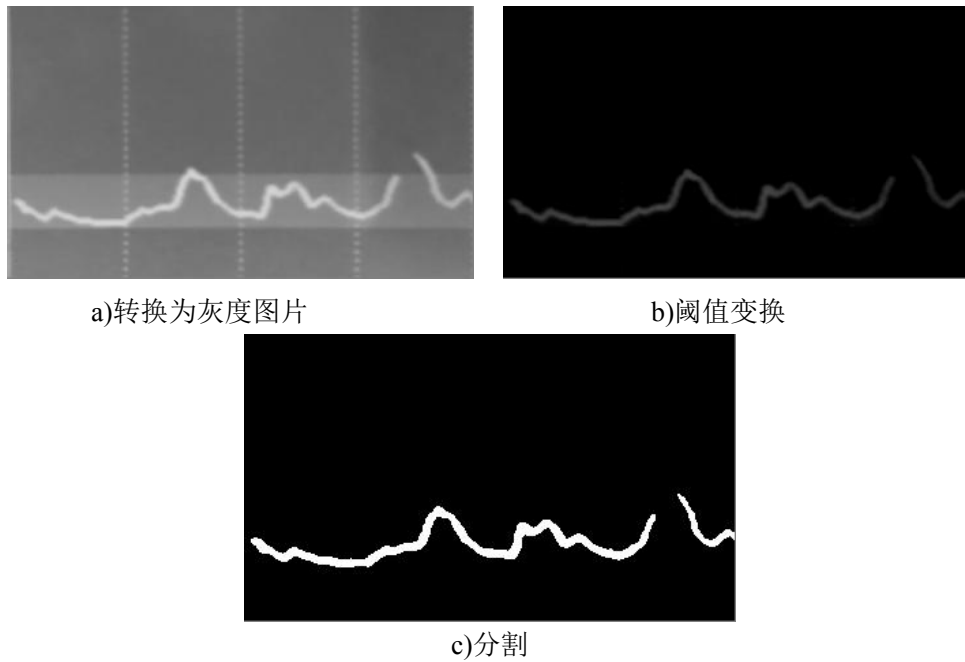


图 4-8 读取曲线数据

4.2.2 图像细化与血糖值计算

我们可以看到由于曲线图像有一定宽度，所以分割得到的曲线图像还需要进行图像细化处理。本文使用当前应用广泛的 Zhang 细化算法。该算法组要思路是通过判断像素点八邻域的信息进行迭代计算。假设某个像素的八邻域标记名称 P1, P2...P9，其中 P1 位于中心：

P9	P2	P3
P8	P1	P4
P7	P6	P5

图 4-9 八邻域标记名称

P_i 的取值为 0 或 1 两种情况，分别代表背景与前景。迭代包以下两步：

(1)遍历全部前景点，如果像素点满足以下条件则删除：

$$\begin{aligned}
 &2 \leq N(P_1) \leq 6 \\
 &S(P_1) = 1 \\
 &P_2 \times P_4 \times P_6 = 0 \\
 &P_4 \times P_6 \times P_8 = 0
 \end{aligned} \tag{4-2}$$

$N(P_1)$ 代表八邻域中灰度值为 1 的个数。 $S(P_1)$ 计算方式为以 P2, P3, P4...P9,

P2 为顺序，累计该过程中像素从背景变为前景的次数。

(2)遍历全部前景点，如果像素点满足以下条件则删除：

$$\begin{aligned} 2 \leq N(P_1) \leq 6 \\ S(P_1) = 1 \\ P_2 \times P_4 \times P_8 = 0 \\ P_2 \times P_6 \times P_8 = 0 \end{aligned} \quad (4-3)$$

迭代过程需要分别循环计算一次以上两个步骤，删除对应的前景像素点，迭代到没有像素点符合以上条件时，迭代计算完成，得到最后的细化结果，也就实现了曲线图像的细化^[37]。

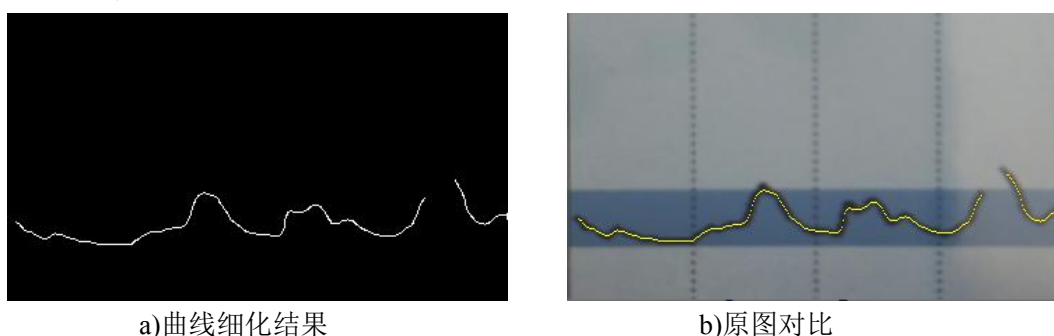


图 4-10 图像细化结果

在得到曲线图像骨架后，可以进行确定图像对应的数据，将每列的前景图像纵坐标作为血糖值，如果某一行有多个前景部分，那么去前景部分的均值作为血糖值。这样的到图像的每列代表的血糖值。

由于需要计算每一分钟的血糖值，而图像宽度小于时间，因此需要进行插值去获得每一分钟对应的血糖值，插值方式选择多种，这里选择简单的线性插值。完整的血糖值计算流程图如下：

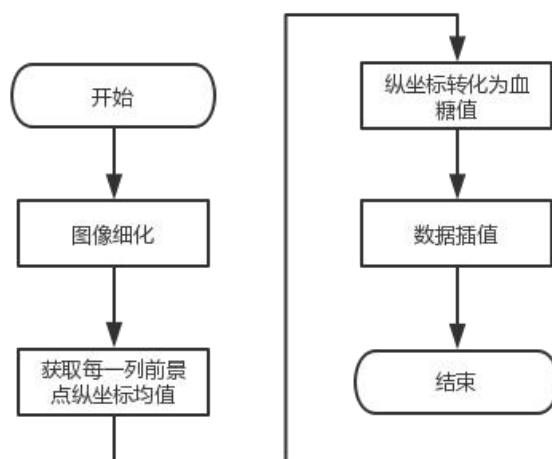


图 4-11 曲线分割图像计算血糖值

最终获得的血糖曲线图像如下：

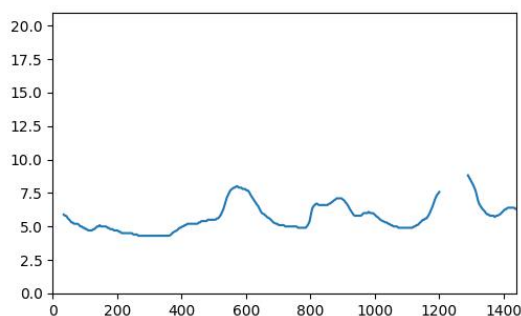


图 4-12 血糖曲线重建图像

4.2.3 算法实验验证

通过对重建的血糖曲线需要与原数据进行对比，检验算法重建结果的误差。已有的对五条样本数据进行重建，通过计算重建数据与样本数据差的绝对值，得到误差数据。

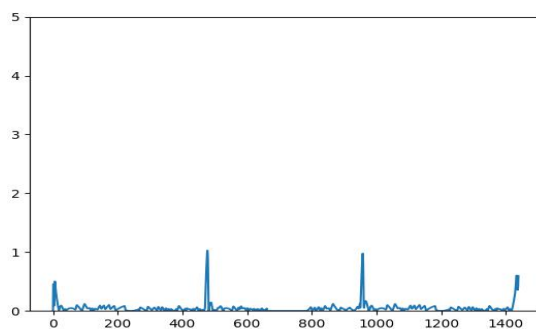


图 4-13 误差图像

对得到的五条误差数据进行统计，最后得到以下结果，其中重建遗漏点代表该时间在原数据上有数值但是重建数据上没有的数值，重建错误点表示原数据上该时刻没有数值但是重建数据该时间有的数值的时间点。

表 4-1 重建数据误差统计

参数	平均数据
平均误差	0.049
最大误差	1.026
平均重建遗漏点个数	23.8
平均重建错误点个数	5.4
误差大于 0.5 点占比	1.39%
误差大于 0.1 点占比	5.49%

可以看到重建数据误差基本小于 0.1，证明该血糖数据视觉检测算法的重建数据误差基本保持在允许范围之内。

4.3 文字数据检测

这里的文字数据检测可以认为是自然场景文本检测识别问题，目前一般处理包含两个步骤：第一步是检测文本，对输入图像检测出文本所在的区域；第二步是将第一步检测道德各个文本区域进行识别，最终识别得到我们想要的文字信息。

随着深度学习技术逐渐发展，近些年来深度学习技术在自然场景文本检测识别也获得了较大的发展，目前逐渐成为主流方法。因此本文主要采用深度学习方法实现文字数据的检测。

4.3.1 文本检测的语义分割网络设计

目前文本检测的普遍做法主要有两种：基于回归的方法和基于分割的方法，由于基于分割的方法能够在像素水平进行预测，故而能更好地描述自然场景中不同形状的文本。因此本文也使用基于分割的方法，借鉴语义分割的深度学习网络进行文本检测。

语义分割是对图像中的每一个像素进行分类，广泛应用于医学图像与无人驾驶等。早期语义分割使用随机森林分类器等方法。在 CNN 大量应用后，深度学习在语义分割上取得巨大突破。从目前的发展趋势看，深度学习方法仍是主导方法。



图 4-14 语义分割示意图

目前语义分割通常使用全卷积神经网络，端到端地学习生成密集的预测图，以指示原始图像中的每个像素是否属于任何文本实例。除此之外，目前语义分割网络架构通常为自编码器结构，包括编码器与解码器两部分^[38]。编码器通常实现对目标的特征提取，由于 CNN 需要较大的感受野，因此在编码过程中使用下采样。而解码器是将编码器在低分辨率提取到的特征从语义上投影到像素空间，获得较高的分辨率，达到像素级分类的目的。通过卷积神经网络实现语义分割时，编码器可以看作是下采样的过程，而解码器可以看作为上采样的过程，这是目前实现语义分割常用的一种结构，通过该结构可以将深的、粗糙的网络层的语义信息和浅的、精细的网络层的表层信息结合起来^[39]。

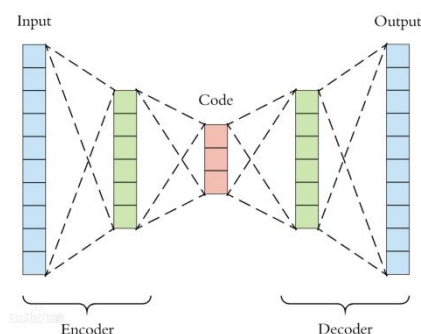


图 4-15 编码器解码器结构

同时由于语义分割需要在像素级进行分类，直接使用低分辨率的特征进行上采样往往获得的结果比较粗糙，因此在解码时除了上采样还会采用跳跃连接结构。不同的架构采用不同的机制例如跳跃连接作为解码机制的一部分。

根据以上常用的语义分割网络结构设计思路，可以设计一个比较简单的卷积神经网络实现语义分割任务，设计的语义分割网络如下图：

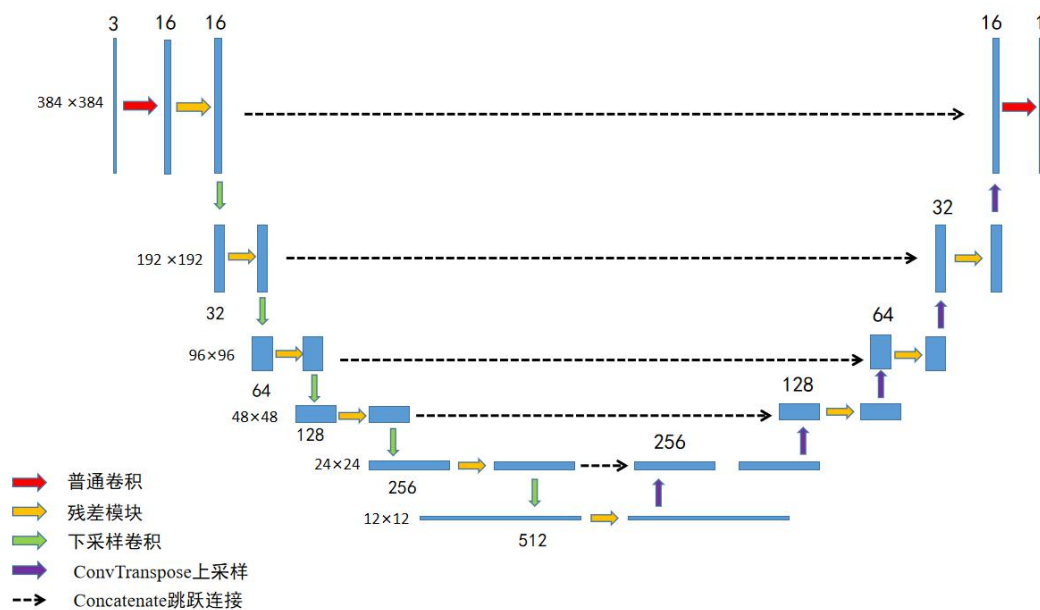


图 4-16 语义分割网络结构设计

设计的结构与 U-Net 比较相似，但是在下采样的处理方式与 U-Net 不同，并且引入了残差模块。整个网络包括以下模块：

(1) 普通卷积：包括卷积层、批标准化、激活函数组成，卷积核参数为卷积核大小为 3，步长为 1，填充为 1；激活函数为 Relu 函数，但是最后一层输出的激活函数为 Sigmoid 函数，这是因为最后是一个二分类问题，文本区域为 1，五文本区域为 0。其他模块大都由卷积模块组成或是基于卷积模块进行修改。

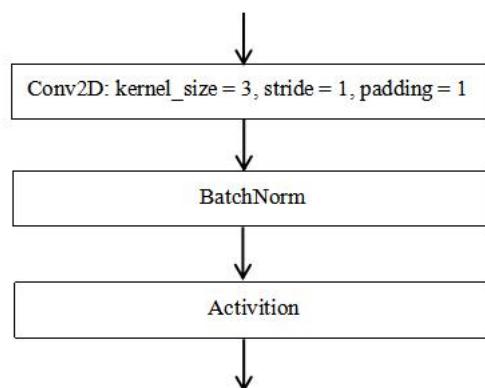


图 4-17 普通卷积模块 Conv 组成

(2) 残差模块：这个部分是借鉴 Resnet 引入的，在普通卷积之后通过引入 Identity mapping，在输入、输出之间建立了一条直接的关联通道，从而使得强大的有参层集中精力学习输入、输出之间的残差，这样有利于网络的收敛。

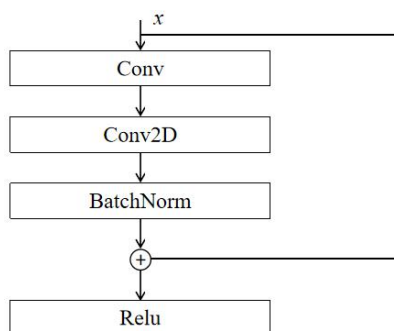


图 4-18 残差模块

(3) 下采样模块：这里下采样不是使用池化层而是使用步长为 2 的卷积层实现，因此这里的下采样模型只需要把普通卷积模块的步长修改为 2。

(4) ConvTranspose 上采样：这里上采样是使用转置卷积 ConvTranspose 实现。转置卷积的计算需要对待转置卷积的特征图进行填充，在特征图相邻的元素之间填充 0，接下来与普通卷积相同，使用卷积核对填充后的特征图进行卷积。

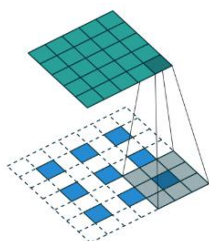


图 4-19 转置卷积

使用转置卷积可以获得更大的输出特征图，并且可以学习转置卷积中卷积核的权重，因此可以通过转置卷积代替插值操作进行特征的上采样。这里将普通卷积中的卷积层改为 ConvTranspose 即可^[40]。

(5) 跳跃连接：将解码器上采样后的特征与原有特征进行融合，这样就保证了不同尺度的特征得到了的融合，从而可以进行多尺度预测。

4.3.2 文本区域检测的训练与后处理

设计好网络后就可以在数据集上进行训练。本文使用的数据集为大规模街景文本(LSVT)数据集，其中包含 30000 个完整的训练数据。对于数据集中具有完整注释的所有图像，标注了文本坐标与文本内容保存在对应的 JSON 文件中。



图 4-20 LSVT 数据集样本

考虑到数据集中的图像大小不同，直接的 `resize` 到统一尺寸会使得图片失真，采用 `letterbox` 方法，将图像长边 `resize` 到指定长度，同时保持图片的长宽比例，最后在短边采用黑色填充，这样输入的图像就可以保持横纵比。

由于分割种类只有文本区域一类，因此训练的损失函数可以选择为二分类交叉熵损失函数，但是考虑到训练中正样本小于负样本，也就是图像中文本区域面积有比非文本区域小，因此选择通过系数增大正样本的损失，减小负样本损失。损失函数公式为：

$$Loss = -3y \log(\hat{y}) - (1 - y) \log(1 - \hat{y}) \quad (4-4)$$

训练网络时选择网络和参数选择如下：

表 4-2 训练参数选择

参数	选择
优化器	Adam 优化器, betas=(0.9, 0.999),
学习率	初始为 0.0001, 每过一个 epoch 降低 0.00002
batchsize	3
MaxEpochs	5

训练过程中的损失函数变化过程如下：

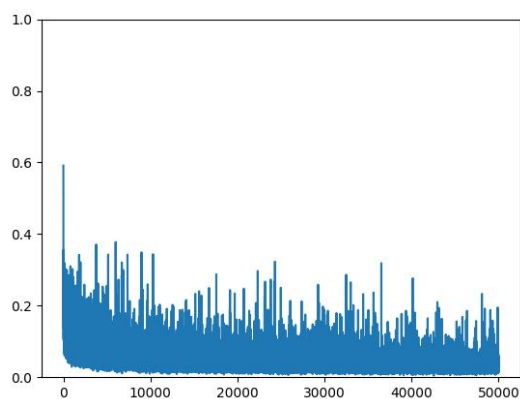
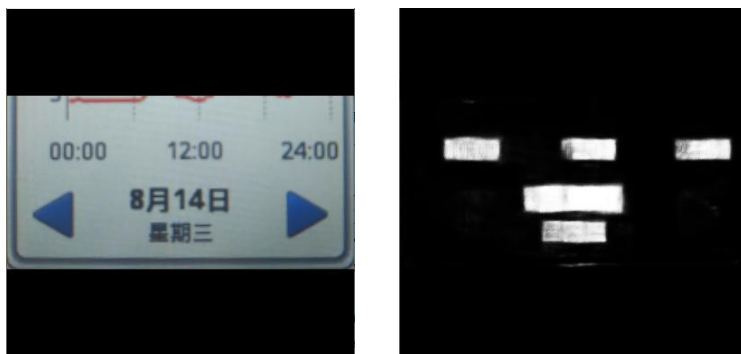


图 4-21 训练中损失函数的变化

最后就得到了分割模型，可以将校正后图像输入到分割网络中检测文本信息。为了提高准确率，这里选择输入图像的下半部分，就可以到文字区域的预测了。



a)输入

b)输出

图 4-22 分割网络的输入与输出

在得到分割网络输出后，还需要进行后处理，包括：

- (1) 对网络输出进行阈值二值化，选择阈值为 0.3；
- (2) 检测二值化图像的区域，在二值化前需要使用形态学开运算去除生成的噪点；
- (3) 对每个区域获得最小外接矩形，将文字区域框选出来；

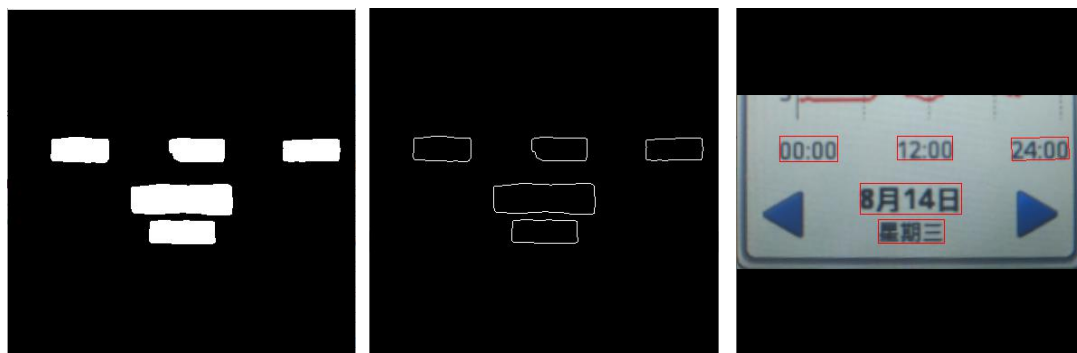


图 4-23 分割网络结果后处理

对分割网络结果进行后处理之后就可以实现了检测文本区域。

4.3.3 CRNN 文字识别

检测出文本区域后，就可以对文本区域图像进行文字识别。本文通过 CRNN 网络对文字进行识别。CRNN 网络主要由 CNN 与 RNN 两个部分组成。CNN 主要是提取图像的特征，RNN 主要是在 CNN 提取的特征的基础上继续提取文字序列特征，最终识别出文本图像对应的文字。

CNN 包括卷积模块、残差模块与下采样模块，各模块与语义分割网络中的模块相同。CNN 的输入为 $48 \times 512 \times 3$ 的彩色文本，通过 CNN 不断进行卷积与下采样提取特征，最终 CNN 输出的特征维度为 $1 \times 32 \times 512$ 。CNN 网络结构设计如下：

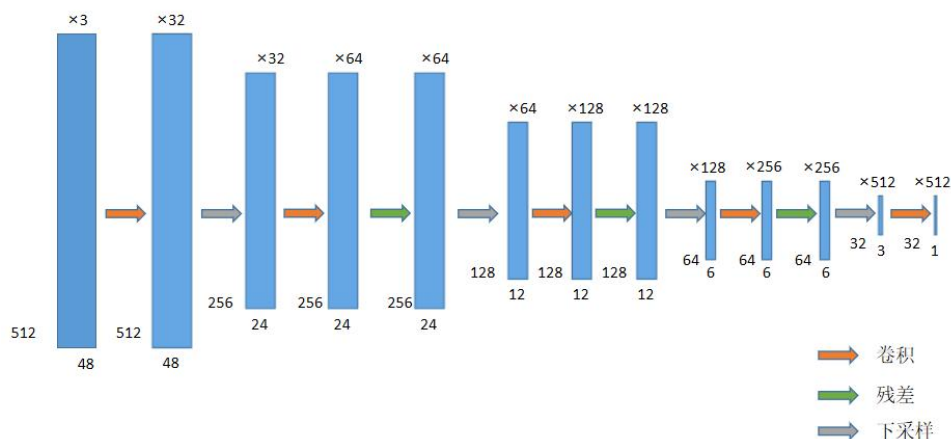


图 4-24 CNN 结构

RNN 由两层 LSTM 组成。由于 CNN 输出的 Feature map 是 $1 \times 32 \times 512$ 大小，所以对于 RNN 最大时间长度 32，即有 32 个时间输入，每个输入列向量维度为 512。对于序列的预测，序列的前向信息和后向信息都有助于序列的预测，所以采用了双向 LSTM 网络，最后将 RNN 输出进行 Softmax，得到最终结果。因为共有 3894 种字符，外加占位符最为一类，共 3895 类，所以最后输出向量维度为 32×3895 ，表示输出每个位置中每个字符的概率。

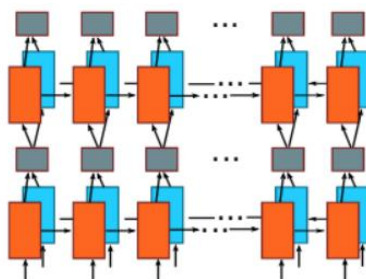


图 4-25 双向 LSTM 结构

组成 RNN 的 LSTM 参数如下：

表 4-3 RNN 参数

层	输入	隐藏层	输出维度
LSTM1	512	256	256
LSTM1	256	256	3895

虽然最后为 32 个位置，但是文本识别是不定长识别，每张文本的长度不固定，输出与输入无法对齐，因此无法用简单的多分类交叉熵损失函数进行训练，这里使用 CTC 损失函数。

CTC 是一种不需要对齐的损失函数计算方法，用于训练网络，被广泛应用于文本行识别和语音识别中。CTC 中非常特殊的一点是引入了一个很特别 blank 标签，用以表示空白，在模型最终匹配路径时需要去掉输出中连续重复的符号和空白。借助 blank 标签，将 LSTM 网络输出的预测 y 做 B 变换，变换公式如下所示：

$$\left. \begin{array}{l} B(-a---p-p-l-e) \\ B(-a---pp-p-l-e) \\ B(-a---p-pl---e) \\ B(-app-----ple-) \end{array} \right\} = apple \quad (4-5)$$

如果字符间存在空位置，则去掉这个空位置，去掉后不合并相同的字符，如果字符间没有空位置，有相同的字符则需要合并。CTC 通过这种多对一的 B 映射将序列解码为最终的类别序列。如果 LSTM 的输出为 x ，那么最终输出 I 的概率为：

$$p(I|x) = \sum_{\pi \in B^{-1}(I)} p(\pi|x) \quad (4-6)$$

对于每条路径 π ，概率为：

$$p(\pi|x) = \prod_{t=1}^N p(t, \pi_t) \quad (4-7)$$

其中 π_t 为路径 π 第 t 位置的类别。CTC 的优化过程基于最大似然原理，因此损失函数为所有正确路径概率的负对数之和：

$$L_{CTC} = -\ln p(I|x) \quad (4-8)$$

由于 CTC 损失函数可微，因此可通过前向-反向动态规划算法有效地计算方程通过不断降低 CTC 损失值达到网络训练的目的，使得预测序列 I' 逐渐逼近真实序列 I 。

这样整个 CRNN 设计完成，整个结构包括：

(1)通过 CNN 自动提取图像特征；

(2)图像卷积特征作为输入，利用双向 LSTM 将卷积特征作为序列，进一步提取文字特征；

(3)对于输出结果与实际标签字符无法对齐的问题，引入 CTC 损失^[41]；

训练采用迁移学习方式，先在大的数据集上预训练，再在小数据集上进行微调。预训练的数据集为公开的中文场景文字识别数据集共包括 21 万张图片，所有文字从场景中拍摄，将拍摄图像中裁剪出文字区域，并将文字区域利用仿射变化，等比映射为一张高为 48 像素的图片。

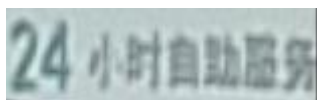


图 4-26 数据集样本

训练网络时选择网络和参数选择如下：

表 4-4 训练参数选择

参数	选择
优化器	Adam 优化器, betas=(0.9, 0.999),
学习率	0.0001
batchsize	50
MaxEpochs	5

训练网络过程中的损失函数变化如下：

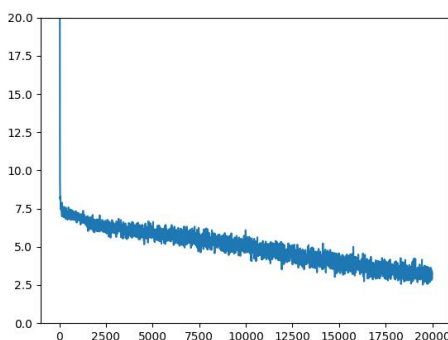


图 4-27 预训练损失函数变化

将得到的预训练模型在小数据集上进行迁移学习微调，小数据集为在屏幕图像中文本区域的样本，通过加噪声、平移、旋转的方式进行数据扩充，共有 1000 张。训练时 MaxEpochs 为 3，剩下参数与与训练相同。训练过程的损失函数变化如下：

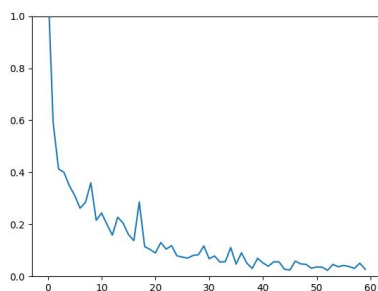


图 4-28 迁移学习损失函数变化

模型训练完成后就可以进行文字的识别，将图像放入模型进行计算，模型的输出还需要通过 B 变换进行解码，就得到了最后识别的文字。



图 4-29 识别效果

4.3.4 算法实验验证

这里主要是对文本区域检测算法与文字识别算法的准确度进行评估。

(1) 文本区域检测

文本区域检测算法准确率评估包括精确率、召回率的计算，两者的计算公式为：

$$presicion = \frac{TP}{TP + FP} \quad (4-9)$$

$$reacll = \frac{TP}{TP + FN} \quad (4-10)$$

精确率代表预测为正样本的数据中真正的正样本比例，召回率代表总的正样本中，模型找回了多少个正样本。

通过对 20 张测试图像的进行测试，最后得到的统计结果如下：

表 4-5 文本区域检测测试结果

正确框选数目	错误框选数目	遗漏数目
91	9	10

因此最终的精确率为 91%，召回率 90.1%

(2) 文字识别

通过对 100 张测试图像的进行测试，正确识别的图像 89 个，识别错误 11 个，因此最终的识别准确率为 89%

4.4 本章小结

本章对校正后的数据进行视觉检测，包括血糖数据与文本数据。血糖数据主要通过传统图像处理方法，首先检测图像中的坐标轴，主要通过检测坐标轴的四个关键点，在检测到关键点后进行进一步校正，对齐坐标，然后通过阈值分割曲线图象，分割出曲线图象后先使用通过图像细化提取曲线骨干，取每一列的均值作为血糖值，再进行插值就获得了时间与血糖值的对应关系。

文本数据检测主要是通过深度学习方法，分为文本区域检测与文字识别两部。

文本区域检测主要是借鉴语义分割的思路，将文字区域作为一类，设计语义分割网络，得到的结果再进行后处理，最终得到文本区域。得到的文本区域作为输入放入文字识别模型，文字识别主要是通过 CRNN 方法，通过 CNN 与 LSTM+CTC 的组合方式进行处理。最终文本区域检测精确率为 91%，召回率 90.1%，文字识别准确率为 89%。

第五章 CGM 血糖仪数据检测系统开发

5.1 引言

糖尿病患者将 CGM 血糖仪数据曲线拍照给医生进行分析是目前远程血糖管理的一种重要方式。开发 CGM 血糖仪曲线识别与数据检测系统主要有以下用途：

- (1) 医生能够使用开发的算法得到定量的血糖数据，进行分析治疗；
- (2) 医生能够更好管理患者的数据信息；

因此在整个算法开发之后还需要进行 CGM 血糖仪图像数据检测系统开发。本章主要通过 Django 框架开发了 Web 端的数据检测系统开发，用户只需在打开网页上传图像，系统就可以自动处理并保存记录。

5.2 总体架构与需求分析

5.2.1 系统总体架构

本系统中的后端框架采用的是 Django。Django 以组件式开发为原则，可以让代码实现最大的复用率。在设计和开发过程中，可以很好地对系统服务进行拆卸、迁移和组装等操作，从而达到缩短程序开发周期，快捷开发一个 Web 应用程序的目的。Django 技术生态中具有许多功能丰富的第三方插件，可以非常容易地集成到现有的应用程序中，极大提高了软件的开发效率。

Django 在架构设计上采用了经典的 MVC 架构设计思路。在具体实现过程中，将原来 MVC 实现为 View 模块和 Template 模块，形成新的 MTV 架构。MTV 架构工作流程如下：

(1) 用户向 Django 服务的 WSGI 网关接口发送 Http 请求，网关接口接收到 Http 请求后，将该请求转化为一个 Http Request 实体对象，然后通过路由服务调用对应的视图；

(2) 视图中的函数在执行过程中从模型层到数据库，首先是根据业务代码调用模型层，模型层再通过 ORM 框架操作底层数据库的数据。

(3) 调用到数据库并完成数据操作之后，Django 框架再将返回结果传给对应的模版层文件进行渲染，最后将渲染结果通过 Http Response 对象返回给浏览器进行展示。

Django 在设计时遵循模块化的设计理念，又把数据库和 Python 的连接做了抽象化设计，以用户数据库为主的模型化技巧让一些第三方网站功能模块也可以轻

松地加入网站，同时在网站的设计中基本不需要使用 SQL 查询语言，而是使用 Python 的方式来处理数据库中的数据，这样既使得查询和处理数据变得简单同时也让数据库的替换或更改变得更加方便^[42]。

Django 以其功能强大并且开发成本低等优点，在众多自动化设备和平台系统得到了广泛的应用，这也是本设计中选用它的理由。

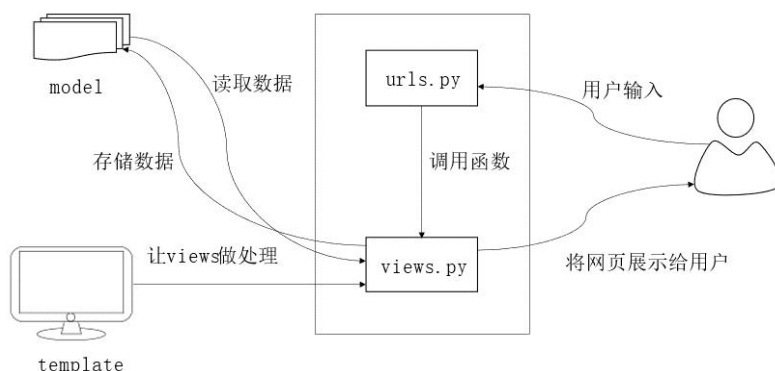


图 5-1 MTV 结构

系统采用 SQLite3 数据库。SQLite3 数据库是一种轻量级、跨平台的关系型数据库，与 Django 配合效果比较好。

5.2.2 系统需求分析

CGM 血糖仪曲线识别与数据读取系统的开发需求包括以下：

- (1)用户模块：主要是用户注册、登录、退出等功能的实现；
- (2)检测模块：对上传图像进行检测，并将检测结果显示给用户，同时能够生成包含检测结果的 Excel 文件，供患者与医生下载查看；
- (3)管理模块：对不同患者用户的信息进行管理以及对用户上传历史记录进行查看管理；

针对上述问题与功能需求，本文开发的 CGM 血糖仪曲线识别与数据读取系统功能框图如下所示：

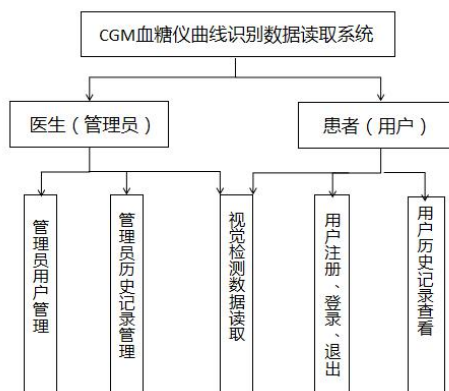


图 5-2 检测系统功能框图

相比而言，管理员与用户基本功能相同，但是管理员权限更高，例如能够删除用户、能够查看所有人历史记录等，而用户只能使用并只查看自己的历史记录等。而使用检测功能两者相同。

5.3 系统模块开发

5.3.1 功能模块设计

根据前一节的 CGM 血糖仪图像数据检测系统开发需求，系统被设计为用户模块、检测模块和管理模块共三个功能模块。

(1) 用户模块

用户模块主要功能包括注册、登录、退出三个部分。用户通过打开浏览器，在界面上点击相关按钮，就可以使用相关功能。

用户点击注册功能时，转到注册页面。用户在注册时需要输入一系列用户信息，例如用户名、密码、邮箱等。后台将验证用户名是否被注册，如果没有注册在检测用户名和密码是否符合规范，若符合就注册成功可以使用。流程图如下：

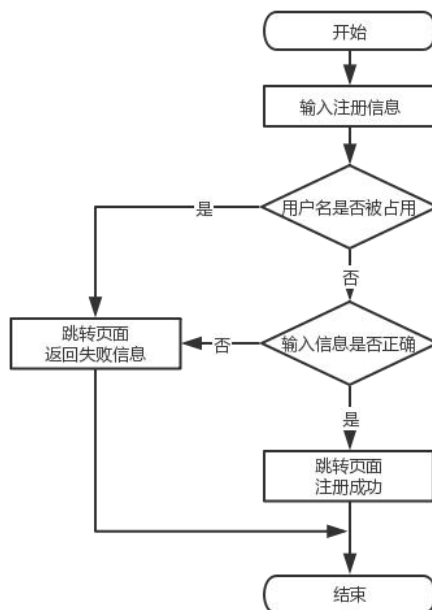


图 5-3 用户注册流程图

用户在页面上输入账号信息。点击登录后，浏览器向系统进行 HTTP 请求，系统后台将进行数据验证，将登录信息与数据库中的用户信息进行比对，通过验证通过后跳转到相应界面，如果有密码错误等错误信息，也需要将错误信息返回到页面。用户登录的流程图如下：

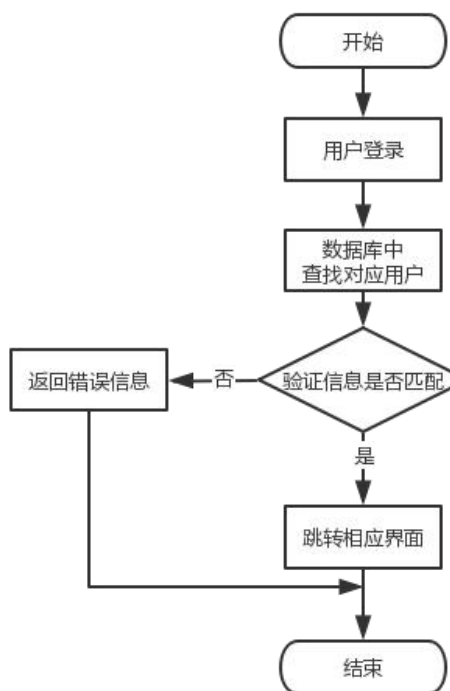


图 5-4 用户登录流程图

(2)检测模块

该检测模块为之前 CGM 屏幕图像视觉检测算法。用户登录之后就可以上传 CGM 血糖曲线图像。在上传界面中，用户选择好待检测的 CGM 血糖曲线图像，通过 POST 请求发送给服务器，服务器保存上传的图像，并自动开始运行视觉检测算法，对 CGM 血糖曲线图像进行检测。检测完成后系统将检测返回到前端显示，用户可以查看检测效果与下载检测的数据。数据主要为包含血糖数据以及其他信息的 Excel 表格。检测模块流程图如图所示。

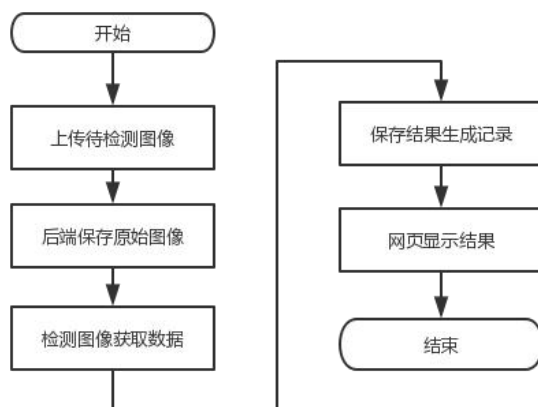


图 5-5 用户登录流程图

(3)管理模块

管理模块主要有两部分，一是能够查看记录，管理员可以查看任何人记录，

但是用户只能查看自己的记录。第二个是用户能够实现对数据的增加删除等，包括增加用户、删除用户、删除记录等。管理模块主要靠 Django 自带的后台功能实现。

除了各功能模块，还需要设计数据库。数据库中主要有用户信息数据表和历史记录数据表。用户数据表中主要有全部用户的 ID、密码、姓名、邮箱、手机号、是否为管理员等信息。历史记录数据表中保存每条历史记录数据，每条数据包括用户名、上传日期、流水号、文件名、上传待检测图像数据信息等。

5.3.2 功能模块实现与测试

(1) 用户模块

打开浏览器输入登录界面对应的 URL，浏览器向系统进行 HTTP 请求转到登录界面，界面包含用户的登录，注册，退出等功能。登录则需要输入正确的 ID 与密码，注册则需要输入符合规则的信息才能注册，首页实现效果如下：



图 5-6 首页效果

登录界面如下：

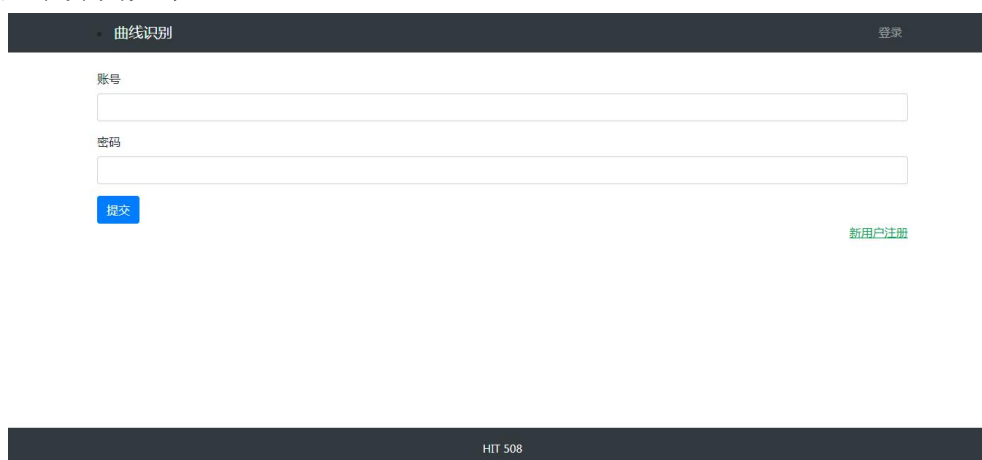
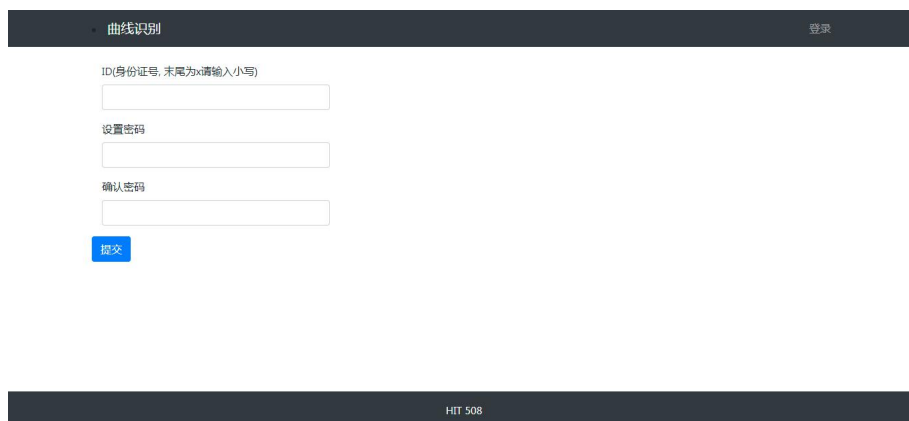


图 5-7 登录界面

注册界面如下：



The registration interface features a dark header with '曲线识别' (Curve Recognition) on the left and '登录' (Login) on the right. The main area contains three input fields: 'ID(身份证号, 末尾为x请输入小写)' (ID (National ID Number, if ending with x, please enter in lowercase)), '设置密码' (Set Password), and '确认密码' (Confirm Password). A blue '提交' (Submit) button is located below the password fields. The footer is dark with 'HIT 508' centered.

图 5-8 注册界面

登录之后效果如下：



The post-login interface has a dark header with '曲线识别' (Curve Recognition) on the left and 'znm 历史记录 退出登录' (znm History Record Logout) on the right. The main area features a large box with the text '请上传图片' (Please upload image). Below this is a file selection area with a '选择文件' (Select file) button and the text '未选择任何文件' (No file selected). Further down are dropdown menus for '类型' (Type) set to '24小时' (24 hours), '拍摄时间' (Shooting time) set to '2021/03/28', and '曲线开始时间' (Curve start time) set to '00:00'. An '上传处理' (Upload and process) button is at the bottom. The footer is dark with 'HIT 508' centered.

图 5-9 登录后首页

(2)检测模块

点击页面中的选择文件，界面显示选择的待检测图像。点击上传处理，则系统开始检测，检测结束后实现检测效果。



The detection module interface shows a dark header with '曲线识别' (Curve Recognition) on the left and 'znm 历史记录 退出登录' (znm History Record Logout) on the right. The main area displays a selected image of a device screen. Below the image is a loading spinner and the text '处理中，请等待' (Processing, please wait). At the bottom are dropdown menus for '类型' (Type) set to '24小时' (24 hours), '拍摄时间' (Shooting time) set to '2021/03/28', and '曲线开始时间' (Curve start time) set to '00:00'. An '上传处理' (Upload and process) button is at the bottom. The footer is dark with 'HIT 508' centered. A status bar at the very bottom shows '正在上传 (12%)...' (Uploading (12%)...).

图 5-10 检测时显示效果

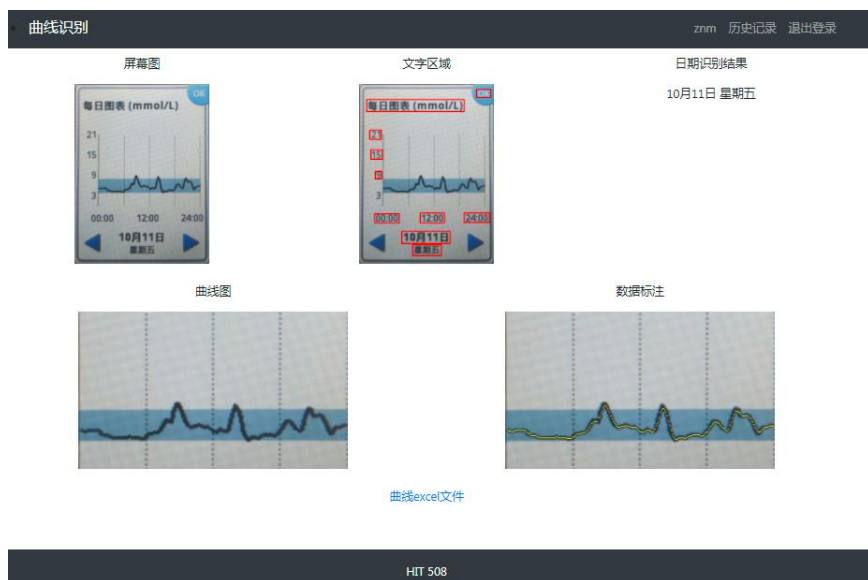


图 5-11 检测效果展示

点击 Excel 文件可以下载到 Excel 文件，里面包含检测的实时血糖数据、图像信息与用户信息。

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
用户名	拍摄日期	上传时间	图片类型	开始时间													
znm	2021-06-18	2021-06-18 0:09:48	24小时	00:00													
原图值																	
x	0		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
y	null	null	null	52	51.5	51.5	51	51	50.5	50	50	51	51	52	52	52	52
计算真值																	
a	0		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
时间	00:00	00:01	00:02	00:03	00:04	00:05	00:06	00:07	00:08	00:09	00:10	00:11	00:12	00:13	00:14	00:15	00:16
血糖值	null	null	null	null	null	null	null	null	null	null	null	null	null	null	5.19375	5.18125	5.16875
识别时间数据	10月11日 星期五																

图 5-12 Excel 数据文件

(3)管理模块

点击右上角的历史记录可以查看每次处理数据的历史记录信息，用户只能查看自己的记录，而管理员可以搜索全部人员记录。



图 5-13 查看历史记录

而增加删除等操作通过 Django 后台实现,主要通过配置 Django 的 admin 模块实现。管理员登录后台可以看到数据信息,包括用户信息与历史数据信息,可以点击页面中相关按钮实现。



图 5-14 登录后台

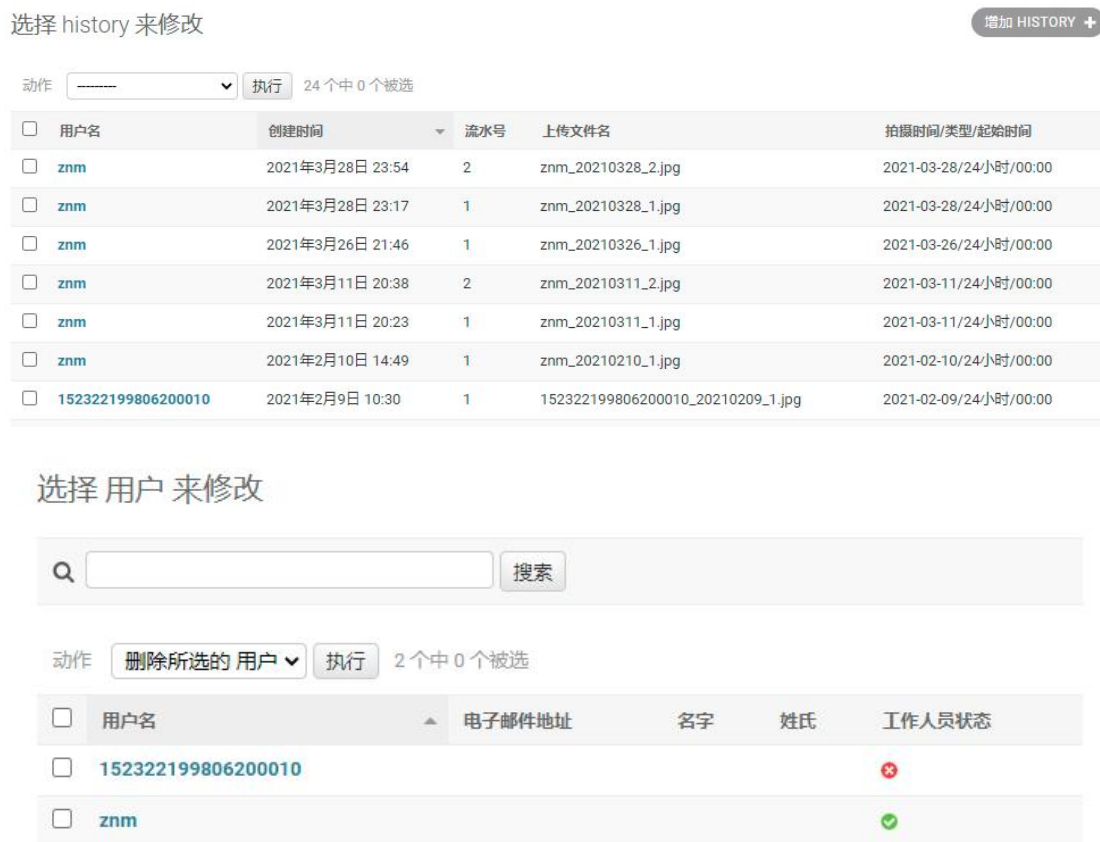


图 5-15 后台增删界面

5.4 本章小结

本章主要通过 Django 框架开发了 Web 端的 CGM 血糖仪图像数据检测系统。系统包括用户模块、检测模块与管理模块。病人可以通过用户模块进行注册，上传 CGM 血糖仪图像，系统就会调用检测模块自动进行检测与保存记录，数据与记录可以通过管理模块进行数据的查看，最终会生成血糖曲线与时间对应的 Excel 表格与图像的处理结果。通过 CGM 血糖仪图像数据检测系统，医生能够使用开发的算法得到定量的血糖数据，并且可以查看历史记录，对于辅助医生进行远程血糖管理有很大帮助。

结论

视觉检测在自动化领域有广泛应用。本文根据糖尿病治疗的现实情况出发,主要研究 CGM 血糖仪屏幕图像的视觉检测,针对 CGM 血糖仪屏幕图像设计了检测算法,包括识别、校正预处理、提取数据三个部分,并最终实现 CGM 曲线识别与数据检测系统开发。本文的主要研究工作和研究成果如下:

(1) 在小样本场景下进行 CGM 血糖仪屏幕图像分类任务,首先使用 HOG 特征结合 SVM 的方式进行研究,在分析 HOG 特征的不足后,提出基于灰度分布的特征对 HOG 特征进行改进,改进后的 HOG 特征既有边缘信息又有灰度信息,并使用改进后的 HOG 特征进行支持向量机分类,最终获得不错的效果。

(2) 研究了 CGM 血糖仪屏幕图像进行校正预处理的研究,包括屏幕图像的分割与投影变换校正两步。对于屏幕图像的分割,提出了基于区域生长的改进 GrabCut 分割算法,通过区域生长算法,计算前景预选框,作为 GrabCut 分割算法的初始条件,可以得到分割效果良好的屏幕图像。对于分割好的屏幕图像,采用投影变换校正,最终得到校正好的屏幕图像可以进行进一步的数据提取。

(3) 针对校正好的屏幕图像,可以进一步视觉检测进行数据提取。提取的数据包括血糖曲线数据与文本数据。对于血糖曲线,采用传统的方法,通过检测坐标关键点进行检测并进一步校正,对得到的坐标图像通过阈值分割获得曲线部分,对得到的曲线部分进行图像细化,并插值计算,最后获得了时间与血糖值的对应关系。对于文本数据,设计语义分割网络进行文本区域检测,对网络输出进行后处理得到文本区域。文本区域作为输入,放入 CRNN 网络就可以获得识别出的文字,CRNN 网络采用 CNN+LSTM+CTC 的方式设计。

(4) 在设计好视觉检测算法后,采用 Django 框架开发了 Web 端的 CGM 血糖仪图像数据检测系统。病人可以进行注册,上传 CGM 血糖仪图像,系统就会调用检测模块自动进行检测与保存记录,数据与记录可以通过管理模块进行数据的查看,最终会生成血糖曲线与时间对应的 Excel 表格与图像的处理结果,医生通过该系统可以帮助病人进行远程医疗血糖管理。

本文设计的视觉算法也具有一定通用性,可以进行修改后应用到身份证、银行卡识别等方面。但是由于时间条件等客观条件限制以及个人能力不足,研究结果还有不足,对于以下问题还有一些改进与提高的空间:

(1) 对于改进 GrabCut 算法,主要受限于边缘检测,对于光照不均匀的情况,无法准确获得屏幕边缘,导致区域生长算法失败。因此可以设计自适应的边缘检

测算法, 优化边缘检测结果, 提高算法成功率。

(2) 在语义分割网络与 CRNN 的训练上, 受限于硬件条件, 最终获得的结果还有提高空间。

(3) 受限于数据集, 基于灰度分布的改进 HOG 特征只进行二分类实验, 对于对分类情况以及泛化能力还有待进一步实验。

(4) 由于处理算法对拍照效果有一定的要求, 因此可以进一步完善检测系统, 在预处理阶段通过对照片的初步分析, 并反馈给用户, 比如建议重新拍照, 提高实用性。

参考文献

- [1] 唐艺. 基于深度学习的连续血糖监测仪故障检测和信号修复研究[D].重庆大学,2019.
- [2] 辛燕,李伟,刘大龙.二甲双胍对糖尿病足溃疡愈合的影响[J].糖尿病新世界,2020,23(02):193-195.
- [3] 宁光. 中国糖尿病防治的现状与展望[J]. 中国科学: 生命科学, 2018, 48: 810-811.
- [4] 张元婷, 王张生, 唐增, 凤玉英. 远程医疗技术在社区 2 型糖尿病患者管理中的应用进展[J]. 健康教育与健康促进, 2019(2):53-54.
- [5] 高希震, 蒋丹玛, 陈碧华. 基于网络的远程医疗在家庭病床 2 型糖尿病患者综合管理中的应用概述[J]. 中外医学研究, 2018(5): 174-175.
- [6] 汪思阳, 沈婷婷等. 互联网远程医疗在糖尿病健康管理中的应用[J]. 上海医药:2020.41(1):26-27.
- [7] 覃群婷, 路文盛, 血糖波动对糖尿病大血管并发症发生机制的研究进展[J]. 山东医药:2017.57(25):110-112.
- [8] 赵亚湘. 指纹图像预处理方法的研究[J]. 微机发展, 1996(4): 40-42
- [9] 冯莉. 原木材积计算机视觉检测系统的研究[D]. 哈尔滨:东北农业大学, 2002.
- [10] 林盘. 指针式仪表视觉检测系统的研究[D]. 西安:西安理工大学, 2002.
- [11] 常辉. 小波变换在套件组装视觉检测中的应用[D]. 武汉:武汉理工大学, 2002.
- [12] 王鹏. 基于机器视觉的钢球表面缺陷检测系统[D]. 哈尔滨:哈尔滨理工大学, 2005.
- [13] 苗世迪. 基于计算机视觉的机械零件尺寸测量研究[D]. 哈尔滨:哈尔滨理工大学, 2006.
- [14] 闵莉,初正恒等. 支持向量机的红外目标自动检测与识别[J]. 沈阳建筑工程学院学报(自然科学版). 2004(01): 75-77
- [15] 李靖. 基于改进的主成分分析类算法的人脸识别[D]. 天津:天津大学, 2002.
- [16] 杨灿. 基于 XGBoost 的遥感图像中道路网络的提取[J]. 微型机与应用. 2017, 36(24):28-30
- [17] 胥海威. 基于改进随机聚类决策森林算法的遥感影像分类研究[D]. 中南大学, 2012.
- [18] 翁秉钧,杨耿杰等. 一种基于改进 K 均值聚类的输电线路覆冰状态侦测方法[J]. 电气技术. 2021, 22(05):43-46

- [19] David E. Rumelhart, Geoffrey E. Hinton, Ronald J. Williams. Learning representations by back-propagating errors[J]. Nature, 1986 (323) : 533–536
- [20] LeCun, Y., Bottou, L., Bengio, Y. and Haffner, P., Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11):2278-2324.
- [21] Krizhevsky A, Sutskever I, Hinton G E. ImageNet classification with deep convolutional neural networks[C]. International Conference on Neural Information Processing Systems. Curran Associates Inc. 2012:1097-1105.
- [22] Simonyan K , Zisserman A . Very Deep Convolutional Networks for Large-Scale Image Recognition[J]. Computer Science, 2014.
- [23] Szegedy C , Liu W , Jia Y , et al. Going Deeper with Convolutions[J]. IEEE Computer Society, 2014.
- [24] He K , Zhang X , Ren S , et al. Deep Residual Learning for Image Recognition[C]// 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2016.
- [25] Long J , Shelhamer E , Darrell T . Fully Convolutional Networks for Semantic Segmentation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2015, 39(4):640-651.
- [26] Ronneberger O , Fischer P , Brox T . U-Net: Convolutional Networks for Biomedical Image Segmentation[C]// International Conference on Medical Image Computing and Computer-Assisted Intervention. Springer International Publishing, 2015.
- [27] Chen L C , Zhu Y , Papandreou G , et al. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation[J]. Springer, Cham, 2018.
- [28] Dalal N , Triggs B . Histograms of Oriented Gradients for Human Detection[C]// IEEE Computer Society Conference on Computer Vision & Pattern Recognition. IEEE, 2005.
- [29]Chapelle, O, Haffner, et al. Support vector machines for histogram-based image classification[J]. Neural Networks, IEEE Transactions on, 1999, 10(5):1055-1064.
- [30] Tartakovsky D M , Broyda S . Vapnik V.N. The nature of statistical learning theory.pdf. 2000.
- [31]杨凯. 基于计算机视觉的无标记手势识别技术研究[D].华北电力大学(北京),2018.
- [32]邹云坤, 杨奎山, 何军等. 基于卷积神经网络和探地雷达的路面缺陷识别技术[J]. 数码设计 (下), 2020,9(6):27-28.
- [33]孙浩荣. 基于 SVM 的食物图像分类算法的研究[D].西安电子科技大学,2016.
- [34]于洋. 基于 CT 影像的早期肺癌计算机辅助诊断关键技术研究[D].东北大

学,2009.

[35]谢畅,钱浩然,徐宝函,周元笙,梅鹄. 一种基于样本增强的手写样本识别方法与系统[P]. 广东省: CN109858488A,2019-06-07.

[36]贾钰峰,哈力木拉提·买买提,冀爽.印刷维吾尔文特征提取之方向码[J].现代计算机(专业版),2011(06):3-5.

[37]方晓蕾. 汽车仪表盘检测系统的字符识别算法研究[D].哈尔滨工业大学,2020.

[38]张菊. 基于浮动车轨迹学习的高分影像城市道路网提取方法研究[D].武汉大学,2019.

[39]林岚,吴玉超,王婧璇,吴水才.基于卷积神经网络的语义分割技术及其在脑神经影像应用中的研究进展[J].北京工业大学学报,2021,47(01):85-92.

[40]雷佳明. 基于语义分割的嵌入式终端水位预警研究[D].华南理工大学,2020.

[41]陈萱,徐美佳,王文涵.基于物联网和图像识别的智慧停车服务平台[J].电脑知识与技术,2020,16(10):187-189.

[42]汤伟.基于 Web 挖掘的个性化视频推荐系统设计与实现[J].电子设计工程,2018,26(18):40-43

攻读硕士学位期间发表的论文及其它成果

哈尔滨工业大学学位论文原创性声明和使用权限

学位论文原创性声明

本人郑重声明：此处所提交的学位论文《基于图像处理的血糖监测数据视觉检测》，是本人在导师指导下，在哈尔滨工业大学攻读学位期间独立进行研究工作所取得的成果，且学位论文中除已标注引用文献的部分外不包含他人完成或已发表的研究成果。对本学位论文的研究工作做出重要贡献的个人和集体，均已在文中以明确方式注明。

作者签名：张诺敏 日期：2021年06月20日

学位论文使用权限

学位论文是研究生在哈尔滨工业大学攻读学位期间完成的成果，知识产权归属哈尔滨工业大学。学位论文的使用权限如下：

(1) 学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文，并向国家图书馆报送学位论文；(2) 学校可以将学位论文部分或全部内容编入有关数据库进行检索和提供相应阅览服务；(3) 研究生毕业后发表与此学位论文研究成果相关的学术论文和其他成果时，应征得导师同意，且第一署名单位为哈尔滨工业大学。

保密论文在保密期内遵守有关保密规定，解密后适用于此使用权限规定。

本人知悉学位论文的使用权限，并将遵守有关规定。

作者签名：张诺敏 日期：2021年06月20日

导师签名：杨旭东 日期：2021年06月20日

致谢

光阴似箭，日月如梭，转眼间两年的研究生生活即将结束。在此期间，学校授课老师们的认真教导，同学们的热情帮助让我在学习中不断前进。在此，我对教导过我的各位老师还有帮助过我的同学们表示真挚的感谢。

感谢我的导师杨旭东老师，为我指明了研究的方向。本论文从课题的选定到论文最终的完成，杨老师都给予了我悉心的指导，并且提出了许多宝贵的意见和建议，再次衷心感谢杨老师对我的指导！

感谢研究生学习期间的各位同学们在生活与学习上对我的帮助。跟你们在一起的经历让我了解到共同协作解决问题的重要所在，感谢与你们在一起的日子，祝各位前程似锦。

感谢学术界的各位前辈们做出的贡献，对你们表示崇高的敬意。

感谢各位评委专家、教授老师在百忙中抽出时间对本文的审阅及指正。