



华南师范大学
SOUTH CHINA NORMAL UNIVERSITY

软件学院
SCHOOL OF SOFTWARE

课程作品设计介绍文档



2023 ~ 2024 学年 第一学期

系统名称: Cars' Info Recognition System

教学单位: 软件学院

课程名称: 数字图像处理

类型: 期末大作业

学号: 20222005089

姓名: 蒋春鸿

2023 年 12 月

目 录

第一章 作品概述	4
一、作品选题	4
二、设计目的及内容	4
1.2.1 作品设计目的	4
1.2.2 作品设计的内容	5
三、软硬件运行环境及开发工具	6
1.3.1 硬件说明	6
1.3.2 软件说明	6
四、作品实现选用材料与资源	6
第二章 整体设计	7
一、作品设计流程图	7
二、各功能板块	8
2.2.1 图像灰度化模块	8
2.2.2 滤波去噪模块	8
2.2.3 阈值分割模块	9
2.2.4 骨架提取模块	9
2.2.5 示数识别模块	10
三、完成作品所用方法及其原理	10
第三章 具体实现过程	11
一、图像灰度化	11
3.1.1 计算机程序框图	11
3.1.2 数学模型与实验原理	11
3.1.3 核心程序	12
3.1.4 处理过程图示	12
二、滤波去噪	14
3.2.1 计算机程序框图	14
3.2.2 实现方法与算法原理	14
3.2.3 核心程序	15
3.2.4 处理过程图示	15
三、阈值分割（即图像二值化）	16
3.3.1 计算机程序框图	16
3.3.2 数学模型与实验原理	16
3.3.3 核心程序	17
3.3.4 处理过程图示	18
四、骨架提取	19
3.4.1 计算机程序框图	19
3.4.2 数学算法模型与实验原理	19
3.4.3 核心程序	20
3.4.4 处理过程图示	20
五、示数识别（※※※）	21

3.5.1 计算机程序框图	21
3.5.2 数学算法模型与实验原理	22
3.5.3 核心程序	27
3.5.4 处理过程图示	31
第四章 实验结果与分析	32
一、实验说明与设置	32
4.1.1 实验准备与图像采集说明	32
4.1.2 实验预期效果	32
4.1.4 实验对照组设置	32
二、实验结果分析	33
4.2.1 基本处理组（空白组）	33
4.2.2 镜头近方向盘（仰角 45° ）	33
4.2.3 镜头远方向盘（俯角 45° ）	33
4.2.4 光线不足组（暗光源）	34
4.2.5 光源阴影组（局部亮）	34
4.2.5 拍摄模糊组（无抖动）	34
4.2.6 拍摄模糊组（有抖动）	35
4.2.7 指针变宽组（光源式指针）	35
三、结果与数据分析	36
4.3.1 数据分析截图	36
4.3.2 数据分析	38
四、总结分析实验结果与数据	39
五、算法比较与作品优缺点	40
4.5.1 算法比较	40
4.5.2 作品方法优点	40
4.5.3 作品方法缺点	41
第五章 总结与体会	42
一、作品开发过程总结	42
二、课题仍存在问题	42
三、技术难点与解决办法	42
四、进一步开发的见解与建议	43
五、收获与体会	44
第六章 致谢	45
第七章 参考文献	47
附录	48
一、图像素材库	48
二、代码架构文件包	48
三、程序运行操作步骤	49

第一章 作品概述

一、作品选题

《数字图像处理基础》期末试卷第一题：实现对指针式汽车仪表盘数据分析

提出一种操作简便的方法，可自动识别出图像中所显示的汽车发动机转速和当前车速信息，并输出在命令行窗口上。

■ 具体要求：

- 1) 自行收集指针式汽车仪表盘图片
- 2) 能够自动识别出汽车发动机转速：xxx 转/分
- 3) 能够自动识别出车速信息：xxx 公里/小时
- 4) 至少提供 10 张不同指示数值、不同形状仪表图片识别结果

■ 本题目所用到的知识点：

彩色变换中的彩色图像灰度化（课本 P233-236）、非线性平滑滤波图像增强方法——中值滤波法（课本 P68-70）、基于阈值的图像分割（即图像二值化）（课本 P186-189）、二值图像中的形态学处理中的骨架提取（课本 P265-266）、基于图像边缘检测的图像分割中的 Hough 变换（课本 P182-185）等。

二、设计目的及内容

1.2.1 作品设计目的

通过设计相关程序代码，实现读取汽车发动机转速和车速的数值的效果，可用于司机的疲劳检测驾驶研究、拓展到指针仪表示数读取等用途，具有广阔的应用前景。

具体而言，一则，在生活中，不少网约车司机经常疲劳驾驶，超速驾驶等危害道路和行车安全等现象更屡见不鲜。将本系统结合 DALLAS 半导体公司推出的 DS18B20 温度传感器和 PulseSensor 光电反射式模拟传感器部署成车载系统，测量温度并简单编程实现 9 位数值读取，便可在通过检测驾驶员心率分析是否疲劳驾驶的同时，利用本系统的基本功能读取实时的车速信息，判断当前司机是否超速行驶。

二则，多年以来仪表识别的难点一直存在。“视觉抄表”，即摄像直读抄表，是一种通过手机或终端设备对水电气表拍照后利用图像识别算法将仪表照片自动识别为读

数的智能抄表方案，具有使用范围广、安装简单、有图有真相、易于使用等特点。仪表表盘图像识别算法是视觉抄表中至关重要的一环，早在 21 世纪初，便有不少专家学者开始从事这一研究工作。快速赋能离线表计，让表网数据更完整，有图有真相的特点彻底解决了买卖双方信用纠纷问题，让决策更可信。相比于人类的视觉，机器视觉不会出现疲劳现象，而且机器视觉识别的精度可以比人眼的识别的精度还要高一些。如今，在存量市场有绝对优势的视觉抄表方案，毋庸置疑成为了仪表智能化 2.0 时代不可或缺的产物。故，本系统未来可以进一步完善，如在仪表上外挂式安装拍照采集设备，设置采集终端定期启动拍照，图像通过无线网络上传至服务端，通过本程序的代码将照片读数转化成数值结果，实现远程抄表、数据分析、收费管理等上层应用服务。



图 1-2-1 总体方案

1.2.2 作品设计的内容



图 1-2-2 指针式仪表盘具体结构示例

如上图，指针式仪表盘，左侧仪表盘为转速表，指针指示数值 $\times 1000$ 表示当前转速（单位：r/min，转/分钟），右侧仪表盘为时速表，指针指示数值为当前车速（单位：km/h，千米/小时）。

- A. 采集汽车仪表盘图像，设置多组对照实验组
- B. 彩色图像进行灰度化及二值化处理（含分量法，中值滤波降噪，图像二值化翻转，去噪填充孔洞等处理）
- C. 利用骨架特征提取，提取图像的骨架特征。
- D. 利用图像分割中的 Hough 变换，读取并展示读数结果。
- F. 反复测试与完善程序，分析实验结果并撰写作品文档。

三、软硬件运行环境及开发工具

1.3.1 硬件说明

图像采集拍摄器械：华为手机 HONOR 30 Lite 3.0.0，型号代码 MXW-AN00

仪表盘实拍图片所在汽车装备：别克 E5 五座 SUV，魏牌蓝山 DHT-PHEV

实拍图片驾车拍摄于广州市官洲隧道、新港西路中山大学北门牌坊

代码编写与程序运行设备：联想系列笔记本

1.3.2 软件说明

操作环境：WIN 11 操作系统

代码编写环境：MATLAB R2019a

流程图绘制环境：Visio 2019

文档写作环境：WPS Word，TeXworks 0.6.6

代码托管环境：Github

四、作品实现选用材料与资源

拍摄材料：汽车，适宜天气，适宜路况

拍摄场地：驾车拍摄，包括且不限于广州市官洲隧道、新港西路中山大学北门牌坊附近，保证安全前提下拍摄

相关函数：Matlab 语言自带的函数库

第二章 整体设计

一、作品设计流程图

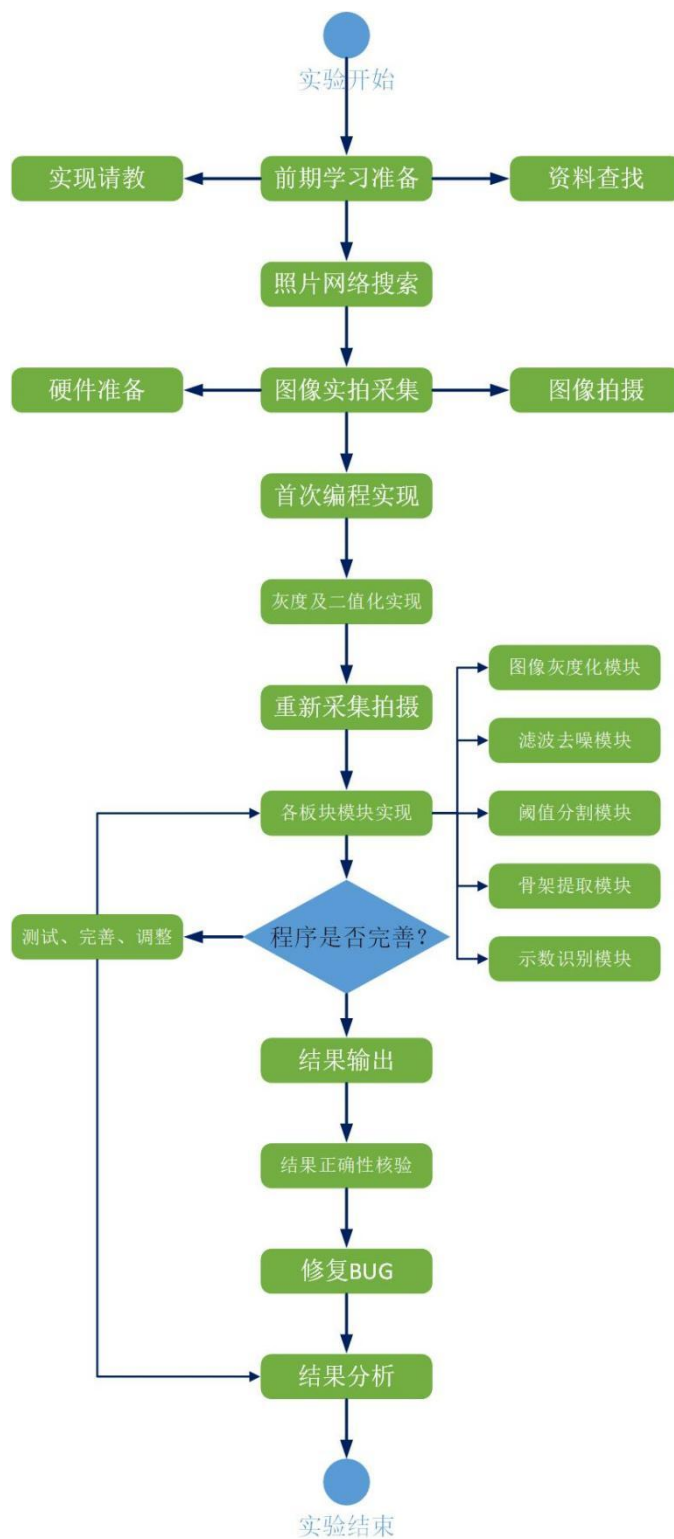


图 2-1-1 作品设计流程图

二、各功能板块

2.2.1 图像灰度化模块

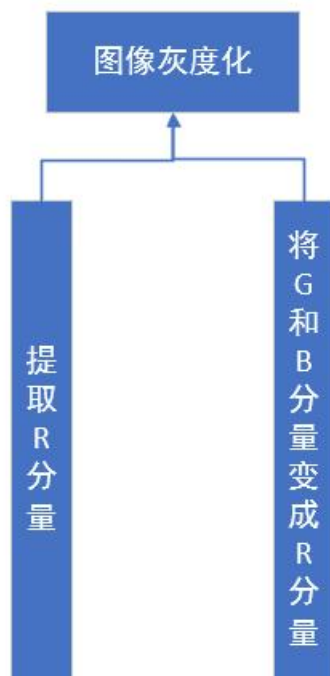


图 2-2-1 图像灰度化模块设计图

2.2.2 滤波去噪模块



图 2-2-2 滤波去噪模块设计图

2.2.3 阈值分割模块

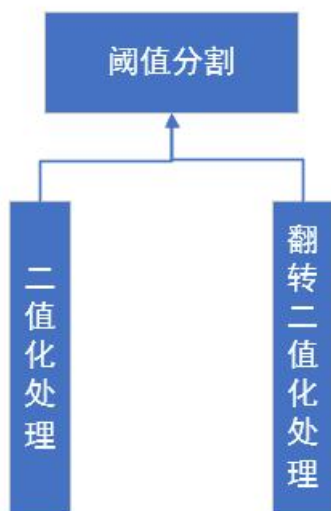


图 2-2-3 阈值分割模块设计图

2.2.4 骨架提取模块

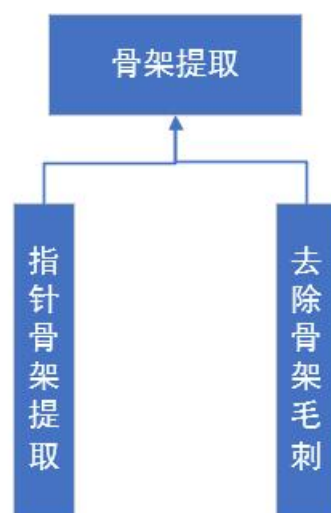


图 2-2-4 骨架提取模块设计图

2.2.5 示数识别模块

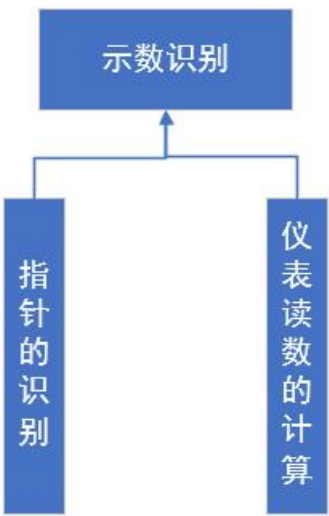


图 2-2-5 示数识别模块设计图

三、完成作品所用方法及其原理

图像灰度化	滤波去噪	阈值分割	骨架提取	示数识别
img(:, :, 1)	中值滤波处理 medfilt2()	二值化处理 im2bw()	形态学处理 bwmorph (‘skeleton’)	产生霍夫变换 hough()
		反转二值图像 bwareaopen()	形态学处理 bwmorph (‘spur’)	寻找变换的峰值数 houghpeaks()
				寻找有意义直线/线段 houghlines()

表 2-3-1 完成作品所用方法及其原理

第三章 具体实现过程

一、图像灰度化

3.1.1 计算机程序框图



图 3-1-1 彩色图像预处理计算机程序框图

3.1.2 数学模型与实验原理

鉴于在图像采集的过程中，会受到许多的干扰，包括自身的和外界的，例如，**外界光照条件的影响，摄像头像素的高低，仪表表盘的清洁程度以及各种振动及噪声的干扰。**受到诸多的干扰后得到的照片可能会是模糊的，可能无法获取我们需要的信息，要想得到有用的信息就必须想办法去掉这些干扰。结果读取和分析之前，需要对图像进行预处理，避免因图像欠曝、光线不足、主体识别颠倒等影响到计数效果。

我们通过摄像头的到的图像都是彩色的，里面蕴含了大量的信息，处理起来相当繁琐，而且计算也非常复杂，容易出现错误，所以为了处理简单，必须要把彩色图片转化为蕴含信息比较少的灰度图片。这种做法有充足的科学依据，《数字图像处理（第4版）》教材 P233 页中明确写道：“**在汽车牌照识别、人脸目标检测及运动目标的跟踪、检测等实际应用中，为了加快图像处理的速度，需要将利用采集设备得到的彩色图像转换为灰度图像。将彩色图像转变为灰度图像的处理称为彩色图像的灰度化处理。**”课本上介绍了三种方法：

1. 最大值法：将输入图像中的每个像素的 R、G、B 分量中的最大者同时赋给输出图像中对应像素的 R、G、B 分量，公式表示为：

$$g_R(x, y) = g_G(x, y) = g_B(x, y) = \max(f_R(x, y), f_G(x, y), f_B(x, y))$$

2. 平均值法：将输入图像中的每个像素的 R、G、B 分量的算数平均值同时赋给输出图像中对应像素的 R、G、B 分量，公式表示为：

$$g_R(x, y) = g_G(x, y) = g_B(x, y) = (f_R(x, y) + f_G(x, y) + f_B(x, y)) / 3$$

3. 加权平均值法：输入图像中的每个像素的 R、G、B 分量的算数平均值同时赋

给输出图像中对应像素的 R、G、B 分量，公式表示为：

$$g_R(x, y) = g_G(x, y) = g_B(x, y) = \omega_R f_R(x, y) + \omega_G f_G(x, y) + \omega_B f_B(x, y)$$

其中， $\omega_R = 0.299$ ， $\omega_G = 0.587$ ， $\omega_B = 0.114$ 最为合理。

这里，我们通过查找资料，学习并使用第四种课本没有的方法：分量法。

4. **分量法**：输入图像中的每个像素的 R、G、B 分量的亮度作为输出图像三个灰度分量的灰度值，可根据应用需要选取一种分量。这里选择**红色**分量，公式表示为：

$$g_R(x, y) = g_G(x, y) = g_B(x, y) = \omega_R f_R(x, y) + \omega_G f_G(x, y) + \omega_B f_B(x, y)$$

3.1.3 核心程序

```
redimage = img(:,:,1); %图像灰度化
figure();
imshow(redimage);
title('R 分量图像');
```

3.1.4 处理过程图示

由于需要展示对光照条件变化等外部因素的图像处理效果，故需要展示原始图像和灰度化后图像，通过控制变量以及对比的方式，观察处理过程是否具有普适性且达到该有的效果。

以下所有的处理过程图示，均采用 img 素材图片中的“测试 3 实拍. jpg”图片进行检验，该图片是别克车型的实拍图片。

如下图所示，对比原图可见，经过预处理之后，两者图像均由彩色图变成灰度图，R、G、B 分量相等，成功呈现黑白效果，不会出现仍有彩色图像影响后续实验步骤的情况。





组图 3-1-2 图像灰度化

二、滤波去噪

3.2.1 计算机程序框图



图 3-2-1 滤波去噪计算机程序框图

3.2.2 实现方法与算法原理

图像在采集、获取和传输的过程中，几乎所有的图像都会收到不同程度的噪声污染，这些噪声有的是可见的，有的是不可见的，有的噪声污染在图像上体现的很明显，有的则是难以察觉的。对于不同的地点，信噪比是不同的，**信噪比越大，图像的质量则越好**，但当信噪比低于一定的数值时，**噪声将会变成人眼可见的颗粒状**，图片就会变得模糊起来，图片质量也会相应的下降。更重要的是，噪声所产生的颗粒有可能会掩盖我们需要的信息细节。噪声污染是不可避免的，而且噪声污染体现在视觉上也是比较明显的，所以必须要对图像进行滤波去噪的处理。

《数字图像处理（第4版）》教材 P64 页中明确写道：“图像噪声在视觉上通常与它们相邻的像素明显不同，表现形式为黑区域上的白点或白区域上的黑点，影响了图像的视觉效果和后续的目标提取等工作，因此需要预先对图像中的噪声进行平滑滤波。取出噪声后的图像会更清晰、明了。”这里使用课本上介绍的**中值滤波处理**方法。这是一种基于排序统计理论的非线性滤波方法，把数字图像中一点的值用该点的一个邻域中各点值的中值来代替，让与周围像素的灰度值相差比较大的像素改为与周围的像素值接近的取值，从而可以消除孤立的噪声点。

基本原理是一种非线性数字滤波器技术，基于某种中值滤波窗口（类似于模板），对原图像中被该窗口覆盖的所有像素的灰度值进行排序，用其中间值代替结果图像中对应于滤波窗口最中间的那个像素的灰度值。由此，把数字图像或数字序列中一点的值，用该点的一个邻域中各点值的中值代替，让周围的像素值接近真实值，从而**消除孤立的噪声点**。其伪代码如下：

表 3-2-1 中值滤波伪代码

中值滤波法

```
Begin
    if ( 确定窗口中心位置像素在原图像上的重合方式 )
        then 在图像上逐像素地移动扫描
            if 对应像素>当前像素
                then 将对应像素值排在起始像素值之后
                    当前索引++
            else
                then 将对应像素值排在起始像素值之前
                    当前索引++
            endif
        endif
        统计像素值个数
        找出位于排序结果中间的那个值
        把找到的中间值赋给结果图像中对应于窗口中心位置的那个像素
    endif
End
```



3.2.3 核心程序

```
redimage=medfilt2(redimage,[7,7]);%中值滤波去噪
figure();
imshow(redimage);
title('中值滤波图像去噪');
```

3.2.4 处理过程图示

如下图所示，呈现的是中值滤波后的图像。可以发现，使用matlab自带函数medfilt2，进行中值滤波处理，去除了部分噪点。特别是在滤除噪声的同时，能够保护信号的边缘，使之不被模糊（优先选用中值滤波而不是均值滤波的原因）。

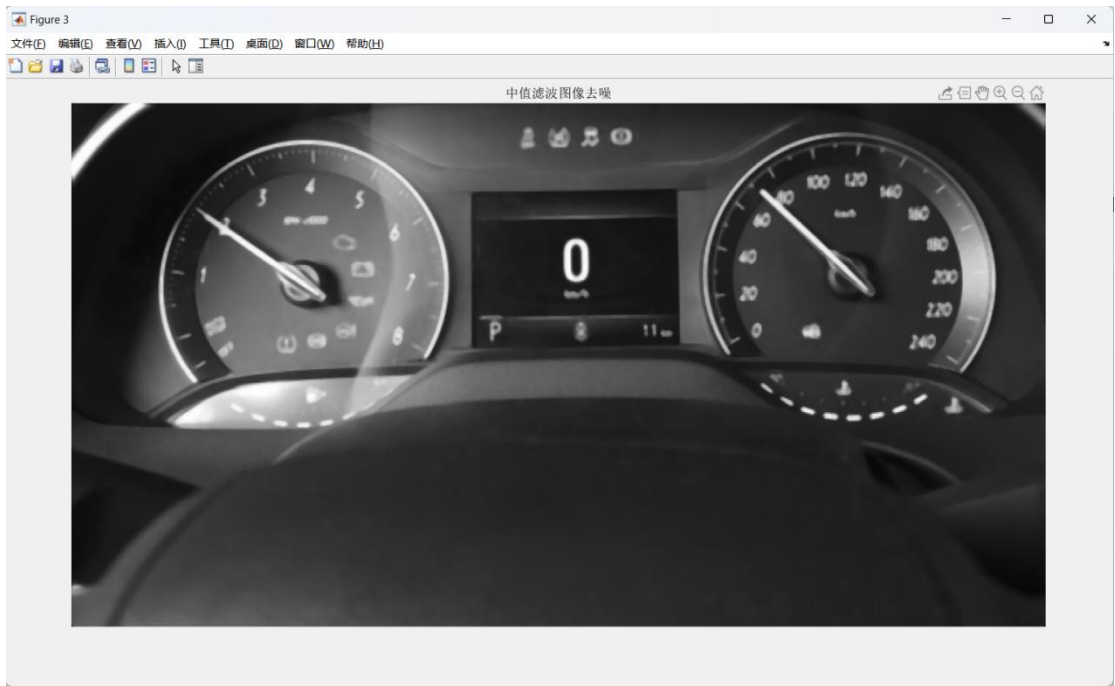


图 3-2-2 图像中值滤波处理后结果图

三、阈值分割（即图像二值化）

3.3.1 计算机程序框图

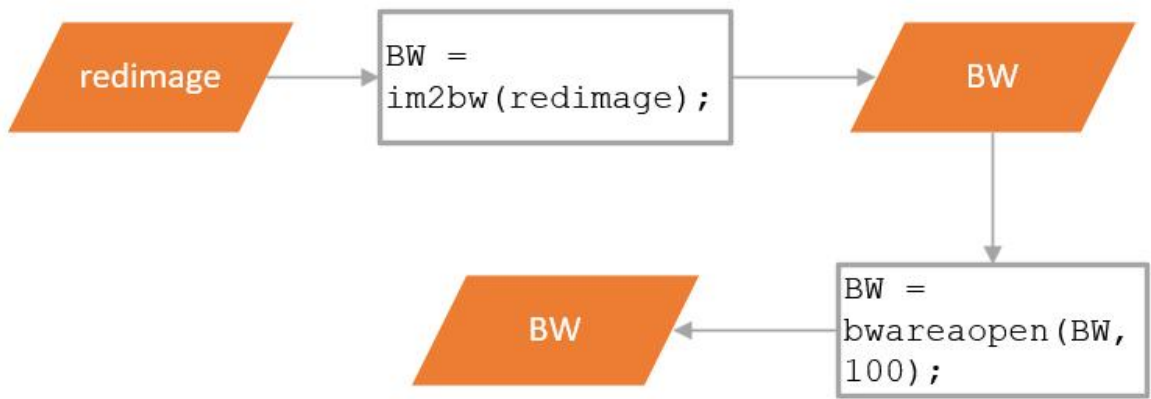


图 3-3-1 阈值分割计算机程序框图

3.3.2 数学模型与实验原理

进行完图像增强后的图像是灰度图像，灰度图像的每个像素都可以在 0-255 之间取值，每个像素都可以有如此多的取值，这样的图像计算和处理起来十分的麻烦，而且容易出错，所以，就要对图像进行二值化处理，使图像的每个像素只能去 0 与 1 两个值，

“0”作为目标区域，“1”作为背景区域，这样计算将变得相对简单，也不容易出错。

本系统中，只有两个仪表盘是要识别的主体，其他如中间的水温表、车辆信息、燃油表、方向盘等都不是本系统用来识别的主体，所以划归为背景板。

用来划分背景区域和目标区域的那个灰度值就称作为阈值，大于阈值的像素值全部设为1，小于阈值的则全部设为0。图像的二值化处理是图像处理中最常用的技术，在图像处理中具有至关重要的地位。主要有三个原因，首先，在彩色图像或者灰度图像的处理过程中，都要先将图像进行二值化处理，然后再进行更深层次的计算。其次，在理论上和方法上图像二值化比灰度图像处理更加系统化，也更加稳定。第三，当图像处理的对象是文字或者图纸的时候，从本质上来说，背景和图形也是一种二值化的图像。图像二值化的主要程序为：

《数字图像处理（第4版）》教材 P186 页中明确写道：“由于图像中的目标与背景有时在灰度上有较大的差异，所以就可利用它们在不同区域上灰度值的不同提取阈值来分割出目标。基于阈值的图像分割就适用于那些物体（前景）与背景在灰度上有较大差异的图像分割问题。”这里包括两个步骤：

1. 二值化处理：

利用 Matlab 自带函数 `im2bw` 函数进行图像二值化处理，得到二值化处理后的 **BW 图像**。因为不指定阈值 `level` 参数的时候，`im2bw` 函数自动将阈值设定为 0.5，所以公式表示为：

$$g(x,y) = \begin{cases} 1 & f(x,y) \geq 0.5 \\ 0 & f(x,y) < 0.5 \end{cases}$$

2. 翻转二值化处理（对于白底/浅色背景图像）：

通过更改 matlab 的自带的 `bwareaopen` 函数得到自生成的**形态学结果** BW 图像。该函数作用是去除因为图像小面积区域，阻止计数值偏大的情况。

3.3.3 核心程序

```
BW = im2bw(redimage); %图像二值化
figure();
imshow(BW);
title('指针二值化结果');
% se = strel('disk',3);
% BW = imopen(BW,se);
```

```
BW = bwareaopen(BW,100); %去除小面积区域  
figure();  
imshow(BW);  
title('形态学结果');
```

3.3.4 处理过程图示

进行二值化处理，同时翻转二值图像，对小面积区域进行去除：

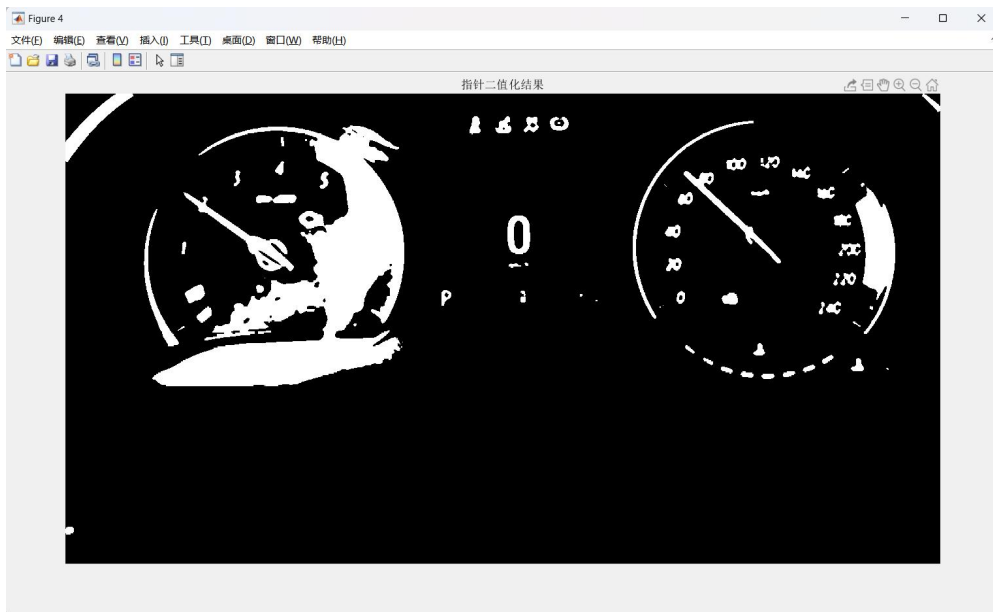


图 3-3-2 图像二值化处理后

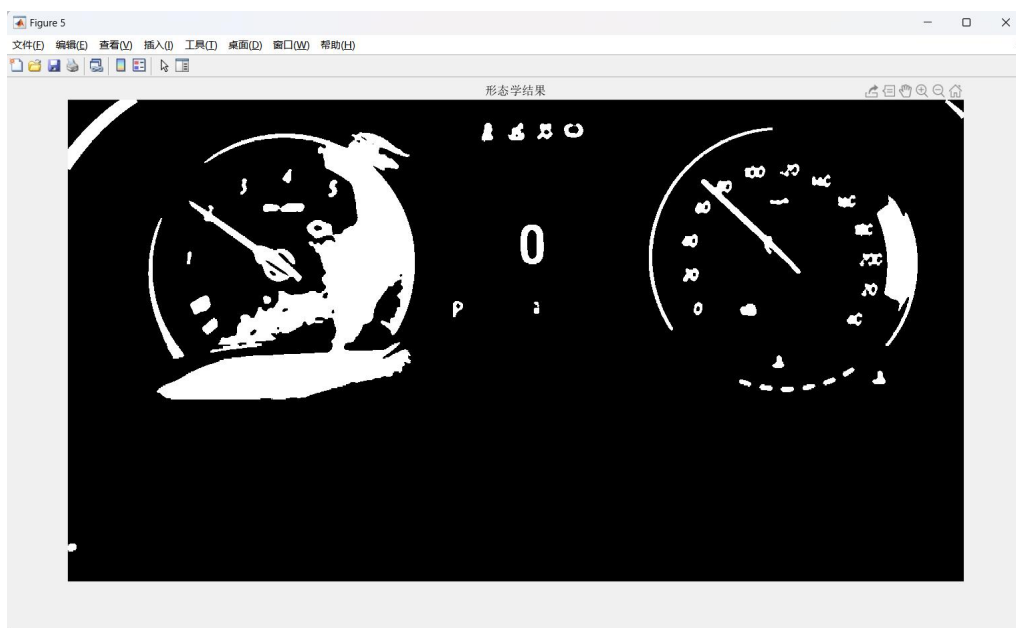


图 3-3-3 去除小面积区域后的形态学结果

四、骨架提取

3.4.1 计算机程序框图

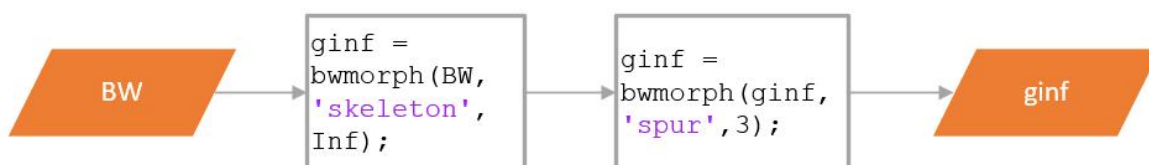


图 3-4-1 骨架提取计算机程序框图

3.4.2 数学算法模型与实验原理

《数字图像处理（第4版）》教材 P1265 页中明确写道：“寻找二值图像的细化结构是图像处理中的一个基本问题，骨架便是这样一种细化结构，并可以用中轴来形象地描述。设想在 $t=0$ 时刻，将目标边界各处同时点燃，火的前沿以匀速向目标内部蔓延，当前沿相交时火焰熄灭，火焰熄灭点的集合就构成了中轴。骨架是图像几何形态的重要拓扑描述，在数字识别、图像压缩编码等方面具有十分广泛的应用。”

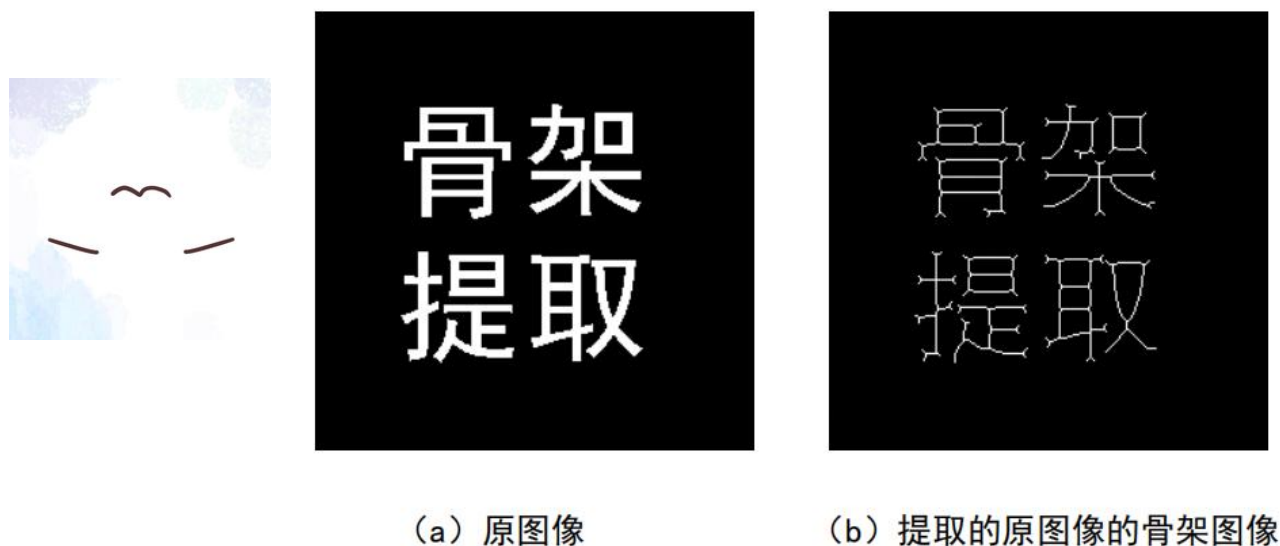


图 3-4-2 骨架特征提取示意图

二值图像 A 的形态骨架可以通过选定合适的结构元素 B ，然后对 A 进行连续腐蚀和开运算来求得。设 $S(A)$ 表示 A 的骨架，则求图像 A 的骨架的过程可以描述为：

$$S(A) = \bigcup_{n=0}^N S_n(A)$$

$$S_n(A) = (A \ominus nB) - [(A \ominus nB) \circ B]$$

其中, $S_n(A)$ 为 A 的第 n 个骨架子集, N 为满足 $(A \ominus nB) \neq \emptyset$ 和 $(A \ominus (n+1)B) = \emptyset$ 的 n 值, 即 N 的大小为将 A 腐蚀成空集的次数减 1。式 $(A \ominus nB)$ 表示连续 n 次用 B 对 A 进行腐蚀, 即

$$(A \ominus nB) = ((\dots(A \ominus B) \ominus B) \ominus B) \ominus \dots) \ominus B$$

由于集合 $(A \ominus nB)$ 与 $(A \ominus nB) \circ B$ 仅在边界的突出点 (如角) 处不同, 所以集合的差 $(A \ominus nB) - [(A \ominus nB) \circ B]$ 仅包含属于骨架的突出边界点。

对完成灰度化、去噪和阈值分割处理后的图像实现骨架提取主要包括两个步骤:

1. 指针骨架提取:

利用 Matlab 自带函数 `bwmorph` 函数进行骨架提取, 带有参数为操作 `'skeleton'`, `Inf`, 即操作无限次, 得到指针骨架 `ginf` 图像。

2. 去除骨架毛刺:

利用 Matlab 自带函数 `bwmorph` 函数进行骨架提取, 带有参数为操作 `'spur'`, `3`, 即操作 3 次, 得到 3 次去除毛刺后的指针骨架 `ginf` 图像。

3.4.3 核心程序

```
ginf = bwmorph(BW, 'skeleton', Inf); %指针骨架提取
ginf = bwmorph(ginf, 'spur', 3); %去除骨架毛刺
figure();
imshow(ginf);
title('指针骨架提取');
[m,n] = size(ginf);
figure();
```

3.4.4 处理过程图示

如下图所示, 可以清晰看到两个仪表盘的线条轮廓:

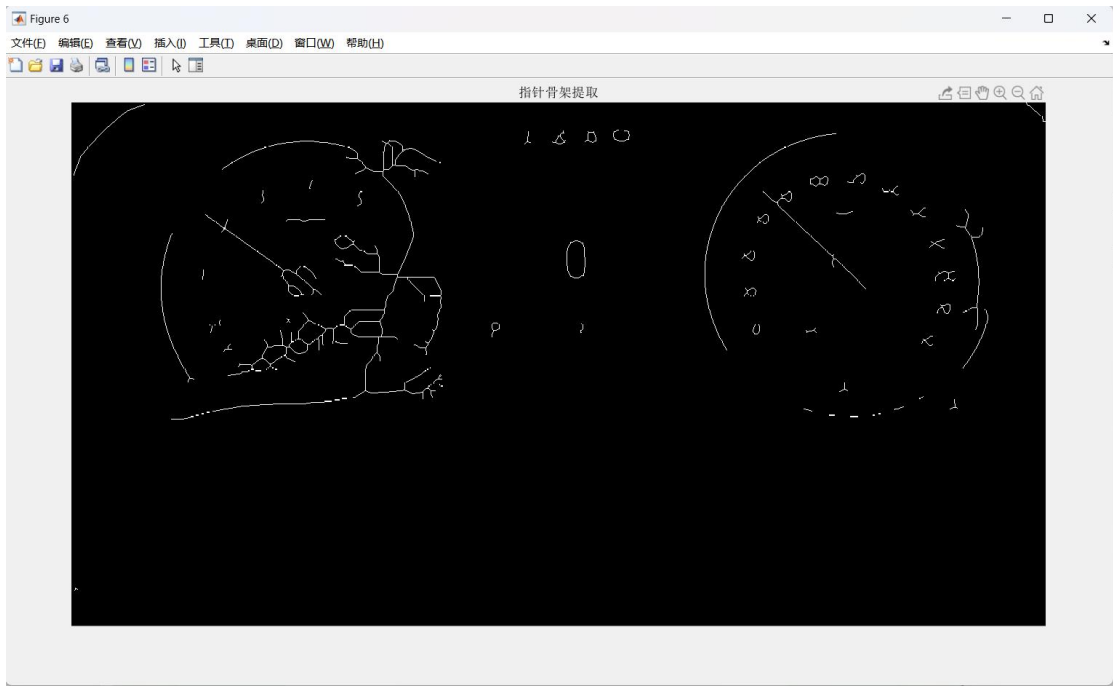


图 3-4-1 指针骨架提取结果显示

五、示数识别（※※※）

示数识别模块是本系统所有模块中最重要的一個模块，也是核心代码模块，所以以下篇幅重点介绍。该模块分为指针的识别和仪表读数的计算。

3.5.1 计算机程序框图

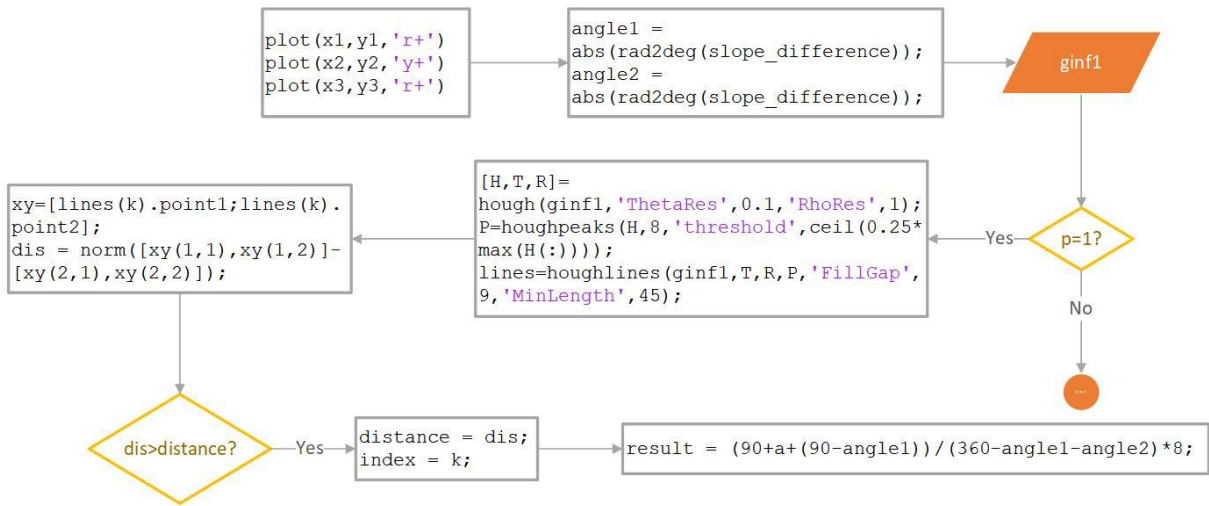


图 3-5-1 示数识别（以转速表为例）计算机程序框图

3.5.2 数学算法模型与实验原理

《数字图像处理（第4版）》教材 P1265 页中明确写道：“霍夫变换（Hough 变换）的基本思想是将图像空间 X-Y 变换到参数空间 P-Q，利用图像空间 X-Y 与参数空间 P-Q 的点-线对偶性，通过将原始图像中给定形状的直线或曲线变换成参数空间的一个点，即把原始图像中给定形状的直线或曲线上的所有点都映射到参数空间的某个点上而形成峰值（点数目累积的值），从而把原始图像中给定形状的直线或曲线的检测问题，转化成寻找变换空间中的峰点问题。”

这意味着我们可以得出两个非常有用的结论：

- 1) 图像空间中的每条直线在参数空间中都对应着单独一个点来表示；
- 2) 图像空间中的直线上任何一部分线段在参数空间对应的是同一个点。

霍夫变换运用两个坐标空间之间的变换，将在一个空间中具有相同形状的曲线或直线映射到另一个坐标空间的一个点上形成峰值，从而把检测任意形状的问题转化为统计峰值问题。从黑白图像中检测直线或者线段是最基本的霍夫变换。接下来使用的算法模型是：使用改进的 Hough 直线检测进行指针的提取。课本上介绍的传统 Hough 直线检测对每个像素点都进行计数和三角函数运算，运算量大，运行速度慢。考虑过程如下：

为提高指针检测的速度，对传统 Hough 直线检测进行改进，根据指针式仪表盘的特征，指针只会落在开始刻度与最大量程刻度夹角以内的区域，可以选择常数 K 确定所占半径比例。

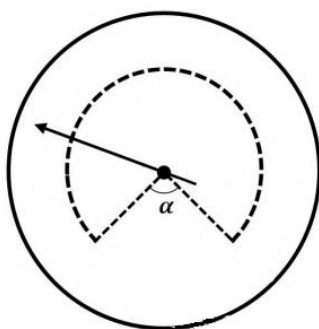


图 3-5-2 选定区域（仪表盘简化）示意图

Hough 圆检测的原半径为 R ，开始刻度与最大量程刻度夹角为 α ， K 为选择圆区域的常数，可求得概率霍夫直线检测区域的面积 S 为下式：

$$S = \frac{(2\pi - \alpha)}{2} \times K^2 R^2$$

以下是示数识别模块的两个步骤：

1. 指针的识别：

指针的识别可以说是识别读数最为重要的环节，仪表的读数都是有指针决定的，指针在采集图像的过程中会连续的变换指向，在摄像头与表盘无相对运动的情况下，摄像头采集到的指针位置不同的两幅图像，**指针也可以看作一条直线**，而且比其他刻度线都要长很多，所以可以采用霍夫变换来检测指针，以及识别其他的刻度。

而寻找最长线段的流程图表示如下：

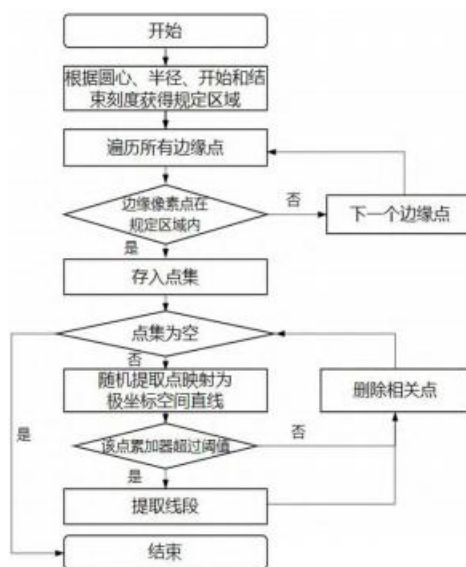


图 3-5-3 直线检测流程图

首先通过霍夫检测检测出所有的直线，包括所有的刻度和指针，而**指针则是所有直线中最长的**，所以，接下来设计程序**求这些所检测出的直线中的最长的线**，然后对直线的起始位置进行标记，并对直线进行标记。

2. 仪表读数的计算：

仪表读数计算使用了角度法，通过计算仪表开始刻度和最大量程刻度之间的角度，以及指针与开始刻度之间的角度，计算指针和开始刻度形成夹角在总量程角度的占比，最后结合仪表的最大量程，计算出仪表的读数，角度法仪表示数计算示意如图所示。



图 3-5-4 角度法仪表示数计算示意图

假设开始刻度为点 A，仪表盘的圆心为点 O，最大量程刻度为点 C，开始刻度与圆心的连线为向量 \overrightarrow{OA} ，指针为向量 \overrightarrow{OB} ，指针与开始刻度所成角度 θ 为式：

$$\theta = \cos^{-1} \frac{\overrightarrow{OA} \cdot \overrightarrow{OB}}{\sqrt{(x_A - x_O)^2 + (y_A - y_O)^2} \times \sqrt{(x_B - x_O)^2 + (y_B - y_O)^2}}$$

开始刻度与圆心的连线斜率为 k_A ：

$$k_A = \frac{y_A - y_O}{x_A - x_O}$$

指针与圆心的连线斜率为 k_B ：

$$k_B = \frac{y_B - y_O}{x_B - x_O}$$

开始刻度与圆心的连线截距为 b_A ：

$$b_A = y_O - k_A \times x_O$$

指针与圆心的连线截距为 b_B ：

$$b_B = y_O - k_B \times x_O$$

记 b_A 与 b_B 的差值 a 为：

$$a = b_A - b_B$$

若 $a \leq 0$ ，则所求角度为 θ ，若 $a > 0$ ，则所求角度为 $\pi - \theta$ 。又仪表盘的最大范围为 MaxRange（转速表的 MaxRange 为 8，时速表的 MaxRange 为 240），则示数 Res 为：

$$Res = \frac{\theta}{\varphi} \times MaxRange$$



现在回到程序中代码。由上述分析可知， a 的大小有两种可能性，这是因为程序读取坐标计算斜率的时候是按照图形平面从左网右扫描的。为了方便描述过程，我分别把这两种可能性叫做从边缘到中心的骨架特征提取和从中心到边缘的骨架特征提取，两者分别对应 $a > 0$ 和 $a \leq 0$ 的情况。

当 $a > 0$ 时，对应指针不超过 y 轴的情况。将图像从左往右扫描，从边缘起始坐标到中心终点坐标，最长的直线的骨架整体是往右下方走的，则程序中计算得出 $angle1$ 和 $angle2$ 的具体含义如下图所示，则 $a - angle1$ 表示指针与 \overrightarrow{OA} 的夹角，则 $360 - a - angle1$ 即为量程的角度（范围从 0 变化到 240）。由此，指针角度占总量程的比例为：

$$\frac{\theta}{\varphi} = \frac{a - angle1}{360 - a - angle1}$$

则有

$$Res = \frac{a - angle1}{360 - a - angle1} \times MaxRange$$

转速表的 MaxRange 为 8，时速表的 MaxRange 为 240。



图 3-5-5 $a > 0$ 情况

当 $a \leq 0$ 时，对应指针超过 y 轴的情况，将图像从左往右扫描，从中心起始坐标到边缘终点坐标，最长的直线的骨架整体是往左上方走的，则程序中计算得出 $angle1$ 和 $angle2$ 的具体含义如下图所示。

（注：一般该种情况只在高速上能出现，但考虑安全性，不敢将车速开到一个

较高的速度，故无法拍摄该情况的图片，只能在上图基础上修改，假想某一时刻开车时时速到了 180km/h，则指针指向右侧仪表盘的 180 的位置。)



a 表示指针与 y 轴的夹角，由数学知识中简单的对称性可知，将 B' 点映射到 B 点方便计算。考虑到 a 是负数，则 $90 + a$ 即为 B' 与 x 坐标轴形成的夹角度数，其对顶角大小也为 $90 + a$ ，则指针与 \overrightarrow{OA} 的夹角为 $90 + a + (90 - angle1)$ ，则 $360 - angle1 - angle2$ 即为量程的角度（范围从 0 变化到 240）。由此，指针角度占总量程的比例为：

$$\frac{\theta}{\varphi} = \frac{90 + a + (90 - angle1)}{360 - angle1 - angle2}$$

则有

$$Res = \frac{90 + a + (90 - angle1)}{360 - angle1 - angle2} \times MaxRange$$

转速表的 $MaxRange$ 为 8，时速表的 $MaxRange$ 为 240。




```

slope_AB = (y2 - y3) / (x2 - x3); % 计算 AB 线段的斜率
slope_BC = (y0 - y2) / (x0 - x2); % 计算 BC 线段的斜率
slope_difference = atan((slope_BC - slope_AB) / (1 +
slope_BC * slope_AB)); % 计算斜率差值
angle2 = abs(rad2deg(slope_difference)); % 将弧度转换为角度

```

```

%% 选择点，确定量程----转速表
fprintf('请选择转速表量程起始点\n');
[x1,y1] = ginput;
hold on;
plot(x1,y1,'r+') %将点在其中标记出来
fprintf('请选择转速表仪表盘中心点\n');
[x2,y2] = ginput;
y0 = y2+60;
x0 = x2+1;
plot(x2,y2,'y+') %将点在其中标记出来
fprintf('请选择转速表量程终止点\n');
[x3,y3] = ginput;
plot(x3,y3,'r+') %将点在其中标记出来
hold off;
% 计算左侧量程初始点与 Y 轴角度
slope_AB = (y2 - y1) / (x2 - x1); % 计算 AB 线段的斜率
slope_BC = (y0 - y2) / (x0 - x2); % 计算 BC 线段的斜率
slope_difference = atan((slope_BC - slope_AB) / (1 +
slope_BC * slope_AB)); % 计算斜率差值
angle1 = abs(rad2deg(slope_difference)); % 将弧度转换为角度
%计算右侧量程结束点与 Y 轴角度
slope_AB = (y2 - y3) / (x2 - x3); % 计算 AB 线段的斜率
slope_BC = (y0 - y2) / (x0 - x2); % 计算 BC 线段的斜率
slope_difference = atan((slope_BC - slope_AB) / (1 +
slope_BC * slope_AB)); % 计算斜率差值
angle2 = abs(rad2deg(slope_difference)); % 将弧度转换为角度
%% 选择点，确定量程----时速表
fprintf('请选择时速表量程起始点\n');

```

```

[x1,y1] = ginput;
hold on;
plot(x1,y1,'r+') %将点在其中标记出来
fprintf('请选择时速表仪表盘中心点\n');
[x2,y2] = ginput;
y0 = y2+60;
x0 = x2+1;
plot(x2,y2,'y+') %将点在其中标记出来
fprintf('请选择时速表量程终止点\n');
[x3,y3] = ginput;
plot(x3,y3,'r+') %将点在其中标记出来
hold off;
% %计算左侧量程初始点与 y 轴角度
slope_AB = (y2 - y1) / (x2 - x1); % 计算 AB 线段的斜率
slope_BC = (y0 - y2) / (x0 - x2); % 计算 BC 线段的斜率
slope_difference = atan((slope_BC - slope_AB) / (1 +
slope_BC * slope_AB)); % 计算斜率差值
angle3 = abs(rad2deg(slope_difference)); % 将弧度转换为角度
%计算右侧量程结束点与 y 轴角度
slope_AB = (y2 - y3) / (x2 - x3); % 计算 AB 线段的斜率
slope_BC = (y0 - y2) / (x0 - x2); % 计算 BC 线段的斜率
slope_difference = atan((slope_BC - slope_AB) / (1 +
slope_BC * slope_AB)); % 计算斜率差值
angle4 = abs(rad2deg(slope_difference)); % 将弧度转换为角度

```

%分左右两部分分别识别

```

for p=1:2
    if p ==1
        ginf1= ginf(1:m,1:round(n/2)-1);
    else
        ginf1= ginf(1:m,round(n/2)-1:n);
    end

    [H,T,R]=

```



```

hough(ginfl, 'ThetaRes', 0.1, 'RhoRes', 1); %hough 变换, ThetaRes
轴变换间隔为 1

P=houghpeaks(H, 8, 'threshold', ceil(0.25*max(H(:))));%
检测投票结果, 制定 8 个特征, 阈值为 0.25*最大值

hold on

lines=houghlines(ginfl, T, R, P, 'FillGap', 9, 'MinLength', 45); %
检测直线, 小于 15, 两直线合并, 检测直线最短长度 102

subplot(1, 2, p), imshow(ginfl, []), title('Hough 变换检测
结果'),
hold on;
distance = 0; %最长距离
index = 1;
%判断指针是哪条线, 可能检测出来很多线, 取最长的线为指针
for k=1:length(lines)
    xy=[lines(k).point1; lines(k).point2];
    dis = norm([xy(1, 1), xy(1, 2)] - [xy(2, 1), xy(2, 2)]); %
线段距离
    if dis > distance
        distance = dis;
        index = k;
    end
    %
plot(xy(:, 1), xy(:, 2), 'LineWidth', 4, 'Color', [.6 .7 .8]);
end
%标记最长线段
xy=[lines(index).point1; lines(index).point2];
plot(xy(:, 1), xy(:, 2), 'LineWidth', 3, 'Color', 'y');
a=[lines(index).theta]; %取出 struct 中 theta
if p == 1 %转速表
    if a < 0
        result =
(90+a+(90-angle1))/(360-angle1-angle2)*8;
    else
        result = (a-angle1)/(360-angle1-angle2)*8;
    end
end

```

```

end
disp(['汽车转速为: ' num2str(result) 'x1000 r/min']);
else
    if a<0
        result =
(90+a+(90-angle3))/(360-angle3-angle4)*240;
    else
        result = (a-angle3)/(360-angle3-angle4)*240;
    end
    if result<0
        result = 0;
    end
    disp(['汽车速度为: ' num2str(result) 'km/h']);
end
end
end

```

3.5.4 处理过程图示

如下图所示，可以清晰看到 Hough 变换的检测结果，其中最长的线段已经被用黄色鲜明地标示了出来：

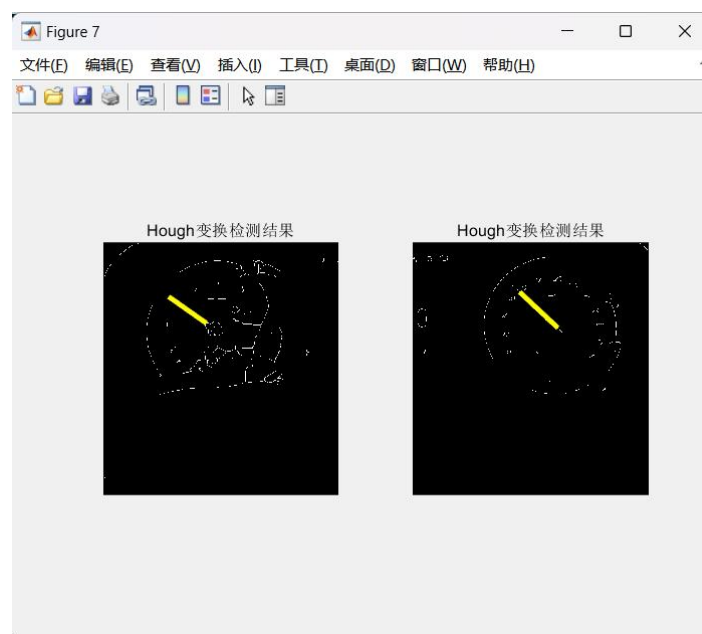


图 3-5-5 Hough 变换检测结果显示

第四章 实验结果与分析

一、实验说明与设置

4.1.1 实验准备与图像采集说明

本实验前期采集了共有 16 张图像，其中网上资源共 12 张，驾车实拍图像共 4 张。筛选了网上资源共 8 张，驾车实拍图像共 2 张，总共实验照片共 10 张，满足题目要求。

4.1.2 实验预期效果

经过局部测试，预测该程序可以对以下图像（环境）进行处理计数：

- A. 仪表盘环境正常的读数
- B. 镜头靠近方向盘的仪表盘读数
- C. 镜头远离方向盘的仪表盘读数
- D. 当车辆进入隧道时，环境变暗光线不足的仪表盘读数
- E. 当车辆驶出隧道时，拍摄有光线导致阴影环境下，仪表盘读数
- F. 拍摄模糊环境下（分有无抖动），仪表盘读数
- G. 非线段式指针而是光源式指针的车辆仪表盘的读数

4.1.4 实验对照组设置

基于以上预期想要实现功能，设置以下几个对照组，如下表所示，对照片进行命名和分组处理：

表 4-1-1 图像分组与命名（对照组设置）

条件/背景色	文件名（文件序号）
基本处理组（空白组）	测试 2、测试 9
镜头近方向盘（仰角 45° ）	测试 1
镜头离方向盘（俯角 45° ）	测试 3 实拍



光线不足组（暗光源）	测试 6
光源阴影组（局部亮）	测试 5
拍摄模糊组（无抖动）	测试 4、测试 10 实拍
拍摄模糊组（有抖动）	测试 7
指针变宽组（光源式指针）	测试 8

二、实验结果分析

4.2.1 基本处理组（空白组）

对测试 2 和测试 9 的读数效果进行了多次测试，其读数**效果达到预期**。

所有的测试结果图示见 4.3.1 节，数据见 4.3.2 节。

总的来说，可见对于在光线良好均匀，拍摄角度垂直于仪表盘所在平面正上方，无阴影，无粘连，拍摄无抖动的图像，可以通过程序计数且效果优异。

4.2.2 镜头近方向盘（仰角 45°）

对测试 1 的读数效果进行了多次测试，其读数**效果达到预期**。

随着倾斜角度的增加，部分仪表盘会存在出现一定程度被方向盘遮挡的情况，导致程序误识别或识别不出仪表盘上最长的指针，同时实验过程中，也发现了由于拍摄对焦影响，将近距离一侧的表盘区域图片定位清晰远距离模糊时，会导致远距离一侧表盘区域的，无法正常识别，导致识别读数结果不正常或是有误。但是如果拍摄倾斜度与拍摄角度在一定的可控范围内时，能够成功计数。

4.2.3 镜头远方向盘（俯角 45°）

对测试 3 实拍的读数效果进行了多次测试，其读数**效果达到预期**。

拍摄过程中倾斜地抬高镜头（大约 45 度）的情况下，程序计数效果一般。虽然会存在当车速较小的时候，转速指针和时速指针被方向盘和驾驶员本人的手部部位遮挡的情况，但是会由于指针还是图片里面最长的线段，虽然看不到 0 刻度线和终点刻度线，

但可以按照间隔和比例直观地感觉出来，从而识别区分了出来。当然也有因为部分颜色较浅且没对焦的情况下，无法被系统识别出来。

总的来说，对于光线正常，拍摄角度不定，镜头有一定倾斜角度，无阴影，拍摄无抖动的图像，可以通过程序进行读数。

4.2.4 光线不足组（暗光源）

对测试 6 的读数效果进行了多次测试，其读数效果整体达到预期。

对于黑暗环境中，拍摄过程中光线不足或仅有暗光源的情况下，程序读数的准确效果会随着图像整体的亮度降低而降低，整体效果一般。反映出对光线不足的图像处理还不足，还有待改善。

总的来说，对于在光线不足或不均匀，拍摄角度垂直于表盘正上方，有一定阴影，拍摄无抖动的图像，可以通过程序计数且效果一般，但并不推荐。同时再一次印证发现识别效果不仅与“其颜色与背景深浅程度是否相近”有关，还与“其颜色在光亮/黑暗环境下是否有反差”有关。

4.2.5 光源阴影组（局部亮）

对测试 5 的读数效果进行了多次测试，其读数效果达到预期。

若图像拍摄过程中因为光源产生阴影或光影的情况下，会导致图像误将阴影识别成一大块区域，但因为骨架提取的过程已经处理了阴影，所以整体差别不大。

同时还通过对此组图像进行裁剪缩放处理后，再次通过程序来进行计数，发现计数效果有所改善。可见识别的更加准确，边缘更加清晰，可能与拍摄光线的影响有关。

4.2.5 拍摄模糊组（无抖动）

对测试 4 和测试 10 实拍的读数效果进行了多次测试，其读数效果达到预期。

对于黑底背景图环境下，拍摄过程中有镜头晃动导致模糊的情况下，在一定的模糊范围内，程序计数效果良好；但是若图像中的表盘表面上光线一般，会出现一定的误差，即程序其实是识别到了这个模糊的指针，但是可能由于图像在计数前的处理中，过度地进行 `bwmorph('skeleton')` 操作，导致产生 `bwmorph('erode')` 的效果，不断腐蚀处理，当二次处理时，被当作了噪点去除，导致读出来的读数值严重偏小。

总的来说，对于在光线良好均匀，拍摄角度垂直于表盘正上方，无阴影，拍摄无抖

动的图像，可以通过程序计数且效果良好。

4.2.6 拍摄模糊组（有抖动）

对测试 7 的读数效果进行了多次测试，其读数效果达到预期。

对于黑底背景图环境下，拍摄过程中有镜头晃动导致模糊的情况下，在一定的模糊范围内，程序计数效果良好，但是若图像过于模糊，镜头抖动的过于严重导致过多重影或是图像过于模糊，仪表的轮廓和指针的整体核心骨架难以区分时，无法识别，系统可能还会直接将其识别成当前表盘上最长的刻度当作读数（可能由于模糊产生了长度上的形变）。

总的来说，对于在光线良好均匀，拍摄角度垂直于表盘正上方，无阴影，拍摄有抖动，图像模糊程度较低的图像，可以通过程序计数且效果良好。

4.2.7 指针变宽组（光源式指针）

对测试 8 的读数效果进行了多次测试，其读数效果达到预期。



图 4-2-1 光源式表盘示例图

这种表盘相对与指针，指示的范围更宽更大，确定斜率和转换角度的时候可能会出现较大的误差。同时，对图像初期的锐化处理以及边缘处理不足也将导致成功率和准确性降低。通过输出的彩色标示图像进行对比不难发现，识别计数的准确性还与拍摄距离有关，由于本组对于表盘拍摄角度较近，图像处理过程中更加精准，排除了一定的误差。所以结果上也更精准，也更加准确的进行光源指示区域正中央中轴线上的“虚拟”指针指示读数。

三、结果与数据分析

4.3.1 数据分析截图

对实验结果进行三次测试，经整理，设计数据分析截图如下：

测试 1.jpg

汽车转速为：1.3477x1000 r/min 汽车速度为：0km/h <i>fx</i> >>	汽车转速为：1.3044x1000 r/min 汽车速度为：0km/h <i>fx</i> >>
汽车转速为：1.3087x1000 r/min 汽车速度为：0km/h <i>fx</i> >>	

测试 2.jpg

汽车转速为：1.2079x1000 r/min 汽车速度为：0km/h <i>fx</i> >>	汽车转速为：1.1975x1000 r/min 汽车速度为：0km/h <i>fx</i> >>
汽车转速为：1.1967x1000 r/min 汽车速度为：0km/h <i>fx</i> >>	

测试 3 实拍.jpg【录视频用】

汽车转速为：2.3547x1000 r/min 汽车速度为：70.2594km/h <i>fx</i> >>	汽车转速为：2.3477x1000 r/min 汽车速度为：69.5193km/h <i>fx</i> >>
汽车转速为：2.3413x1000 r/min 汽车速度为：69.3324km/h <i>fx</i> >>	

测试 4.jpg

汽车转速为：2.2985x1000 r/min 汽车速度为：92.4347km/h <i>fx</i> >>	汽车转速为：2.2772x1000 r/min 汽车速度为：92.3515km/h <i>fx</i> >>
汽车转速为：2.2977x1000 r/min 汽车速度为：94.7784km/h <i>fx</i> >>	

测试 5.jpg

汽车转速为: 0.71973x1000 r/min
汽车速度为: 0.77146km/h

fx >>

汽车转速为: 0.68559x1000 r/min
汽车速度为: 0km/h

fx >>

测试 6.jpg

汽车转速为: 0.81451x1000 r/min
汽车速度为: 0.30256km/h

fx >>

汽车转速为: 0.85534x1000 r/min
汽车速度为: 0km/h

fx >>

测试 7.jpg

汽车转速为: 2.0264x1000 r/min
汽车速度为: 115.2097km/h

fx >>

汽车转速为: 2.006x1000 r/min
汽车速度为: 114.8541km/h

fx >>

测试 8.jpg

汽车转速为: 1.7106x1000 r/min
汽车速度为: 32.4312km/h

fx >>

汽车转速为: 1.7148x1000 r/min
汽车速度为: 32.2104km/h

fx >>

测试 9.jpg 【录视频用】

汽车转速为: 2.0006x1000 r/min
汽车速度为: 78.2888km/h

fx >>

汽车转速为: 2.0275x1000 r/min
汽车速度为: 77.0982km/h

fx >>

测试 10.jpg

汽车转速为: 0.69804x1000 r/min
汽车速度为: 0.56978km/h

fx >>

汽车转速为: 0.82276x1000 r/min
汽车速度为: 0km/h

fx >>

汽车转速为: 2.0315x1000 r/min
汽车速度为: 114.9365km/h

fx >>

汽车转速为: 1.6616x1000 r/min
汽车速度为: 29.9748km/h

fx >>

汽车转速为: 2.0452x1000 r/min
汽车速度为: 78.0474km/h

fx >>



汽车转速为：1.3165x1000 r/min
汽车速度为：40.0441km/h
fx >>

汽车转速为：1.315x1000 r/min
汽车速度为：40.1347km/h
fx >>

汽车转速为：1.3173x1000 r/min
汽车速度为：38.5563km/h
fx >>

表 4-3-2 各对照组计数个数统计表与计数成功率

4.3.2 数据分析

鉴于拍摄以及其余环境因素的影响，综合各项不定因素分析上述实验结果：

表 4-3-1 程序对不同图片的读数统计的结果概率表（以时速表读取为例）

文件名（序号）	第一次读数 /km/h	第二次读数 /km/h	第三次读数 /km/h	前三次平均读数 /km/h	人工读取读数 /km/h	相对误差/%
测试 1	0	0	0	0	0	0
测试 2	0	0	0	0	0	0
测试 3 实拍	70.2594	69.5193	69.3324	69.7037	65.5	6.41
测试 4	92.4347	92.3515	94.7784	93.1882	100	6.81
测试 5	0.77146	0.56978	0	0.44708	0	/
测试 6	0.30256	0	0	0.10085	0	/
测试 7	115.2097	114.9365	114.8541	115.0001	116	0.12
测试 8	32.4312	29.9748	32.2104	31.5388	29	8.75
测试 9	78.2888	78.0474	77.0982	77.8115	80	2.73
测试 10 实拍	40.0441	40.1347	38.5563	39.5784	40	1.05

其中，相对误差=(|前三次平均读数-人工读取读数|)/人工读取读数。

共有 10 张有效图像通过程序进行读数，排除掉无法计算相对误差的两张图片，共



有 8 张，其中成功计数无误的图像共有 2 张，成功率为 25%。

由于测试组别较少，且实验过程中为探索该程序的测试能力范围，测试多种不同情况下程序读数的相对误差，故个别组别相对误差仍然较高或是成功率将近为零。相对误差超过 10% 的图片的图片不放入 imgTest 源图片文件夹。

同理，程序仍然以转速表读取为例，对不同图片的读数统计的结果概率表进行了统计和调查。由于篇幅原因，不在本文档中体现，最终的结果如下：

共有 10 张有效图像通过程序进行读数，除去两组无法统计相对误差的图片，共有 8 张，成功在合理的误差范围内读数的图像共有 5 张，成功率为 62.5%。



四、总结分析实验结果与数据

通过对实验结果与数据分析可以总结得出以下结论：

1. 程序可以实现在光线良好且均匀，拍摄角度垂直于表盘正上方，无阴影，拍摄无抖动的指针图像中进行读数，效果优异，成功率高。

2. 若拍摄过程中因机位抖动，存在一定范围内的模糊（肉眼还是可以看到指针在仪表盘上指示的数值明显）的情况下，程序对于此种情况下计数效果良好。

3. 对于在光线不足或不均匀造成阴影的情况下，同样可以通过程序计数但是计数效果一般，如果因为光源导致阴影产生时，会导致表盘的 Hough 圆边缘区域识别偏大，可以推测出若此种情况下，刻度线拟合存在一定范围内的可接受的误差，导致读数值偏小。

4. 对于拍摄角度不定，镜头有一定倾斜角度，与表盘形成夹角，无阴影，拍摄无抖动的仪表盘图像，可以通过程序进行计数，虽然计数效果一般，随着角度变化时好时坏，但是此效果可期。

5. 极大部分情况下，程序对于方向盘黑色背景图下的仪表盘图计数效果比在有套上保护套的方向盘的背景图下的要好得多。

6. 若图像采集过程中没有对焦到仪表盘指针（局部对焦）时，会导致识别标示有误差。

7. 程序计数的准确性与车内行车时的光线、仪表盘的颜色深浅均有关。表面光线足够适中、表面越光滑、颜色与背景底色反差越大，识别计数效果更好，更准确。

五、算法比较与作品优缺点

4.5.1 算法比较

下面几张表是对程序中出现的部分处理方案与算法思维进行比较：

表 4-5-2 灰度化处理算法思想对比表

处理问题	灰度化处理	
处理方案 / 算法思维	最大值法灰度化	加权平均值法/调用函数
效果	图像过于亮 丢失的图像细节较多	灰度化后过于暗 不易于进行目标对象 与背景区域的区分

表 4-5-2 去噪增强图像（滤波）处理算法思想对比表

处理问题	增强图像去噪（滤波）	
处理方案 / 算法思维	均值滤波	中值滤波
效果	线性滤波算法 不能很好地保护图像细节 在图像去噪的同时 破坏了细节部分 使图像变得模糊 不能很好地去除噪声点	非线性平滑技术 对脉冲噪声 有良好的滤除作用 在滤除噪声的同时 能够保护信号的边缘 使之不被模糊

4.5.2 作品方法优点

通过多次实验以及分析实验结果可以发现本作品方法有以下优点：

1. 本程序的逻辑清晰，处理思路主流。没有使用其他一些目测上很高深的、实际上效果不好的方法进行问题解决，从图像的预处理（灰度化、图像去噪）到阈值分割，再到图像提取，Hough 变换，都是最基本的思想和最主流的方法。哪怕是对原方法进行

创新，都能用最简单的模型，尽量用更简单的原理，力求阐述明晰。

2. 通过中值滤波的方式，在滤除噪声的同时，能够保护信号的边缘，使之不被模糊。

3. 二值化处理过程中，有进行翻转二值化处理的过程，以至于可以处理浅色背景的方向盘遮挡的图像，使得读数标识准确，去除小面积的区域，阻止读数值偏大。

4. 理解了每个方法的原理，但在编程实现的过程中，优先考虑通过 Matlab 自带的库函数，代替了使用原始方法定义底层函数的过程，降低了项目开发的难度，提高了识别的准确率和成功率。

5. 作品处理设置了正常图片的对照，还形成了 7 组以上的对照实验比较，以观察总结作品实现的缺陷，不断完善，同时还总结了图像处理过程中的一些影响因素与结论。

6. 本程序可以部署成车载系统，或者在仪表上外挂式安装拍照采集设备，让项目发挥出更大的价值。

7. 本系统可以实现人机交互，需要用户用鼠标指定起始点、表盘的圆心、量程的结束位置等三个点，两个表盘共六个点。

4.5.3 作品方法缺点

通过多次实验以及分析实验结果可以发现本作品方法有以下缺点：

对于远距离拍摄、光线过度不足、图像过度模糊、拍摄角度不正确的图像处理与计数效果不理想；虽然对转速表的读数有一定的成功率，但对时速表的读数的成功率较低；程序容易受到外界其他环境因素的影响；图像预处理过程时间较长导致整体效率较低；没有制作 GUI 界面，识别出来的读数的结果是打印在了命令行窗口，增加了使用者的操作流程，让流程变得繁琐；读数的误差率较高；数据采集的样本量小，可能还有其他未考虑到的对照组情况，等。

第五章 总结与体会

一、作品开发过程总结

作品前期准备过程中，先是查阅了各项资料，熟读了许多论文，熟悉了相关领域的研究现状，了解学习到曾经其他学者实现过的相关技术，引发思考和优化后着手准备。

图像灰度化过程中，我一开始使用了系统自带函数 `rgb2gray` 直接将 RGB 图像转化为二维灰度图像，为了对图像灰度值进一步调整优化，使用 `imadjust` 函数。使用过程中发现该函数只能使用于二维图像，不得不使用 `rgb2gray` 转为二维图像后才能再进行调整优化。最后经过敲定，选择了课本上没有的更为创新的分量法。

接着是骨架提取，在 `csdn` 上找到了许多表盘分析的文章，但大都是仪表盘而不是汽车表盘，且里面的算法改进过程需要融合多篇博文，在充分理解的基础上设计出了自己的思路。

最后就是根据结果不断反复的调试处理，进行多次腐蚀膨胀处理，改善到最合适的处理模型与对应参数，调整预处理增亮效果，通过部分典型图像的测试，调整与优化代码。

二、课题仍存在问题

验证的图片较少，这样可能会导致部分图片的情形没有考虑到位；

程序不够完善，对于一些特殊场景下的图像处理效果不够理想；过度腐蚀与开运算处理；部分处理过程出现冗余繁杂的情况（灰度化处理）；灰度化处理后图像较暗，与背景区分度还可以更高，算法还可以更优；容错率较低，容易产生误差；实验样本量不足，样本选取代表性不够；程序处理较慢，预处理过程可以适当简化；没有制作 GUI 界面，结果打印在了命令行窗口；不能自动识别量程的起始和终止位置，每投喂一张图片需要用户自己去指定。

三、技术难点与解决办法

表 5-3-1 技术难点与解决办法表

技术难点	解决办法
白底/浅色背景图像二值化结果反转	运用函数 bwareaopen 进行翻转二值化
去除大片阴影区域	更改 bwareaopen 函数进行处理
形状不同，量程不同，图片中表盘的位置不同，那起始读点的位置不同	适应 matlab 特点，运用 plot(x,y,'y+') 函数实现人机交互操作
多个模块的代码编写存在困难	适应 matlab 特点，直接调用 Matlab 内置库函数

四、进一步开发的见解与建议

- 根据存在的问题，提出以下开发见解与建议：
- 1. 进一步完善程序结构，使得程序有一个可视化界面，同时更加智能高效；
 - 2. 增大样本量，选用**更典型样本**进一步调试处理；
 - 3. 针对一些特殊场景下的图像，进一步增加处理过程，实现对应效果，减少误差，降低容错率；
 - 4. 减少并**优化腐蚀处理过程**，适当锐化让图像轮廓更清晰；
 - 5. 简化预处理流程，优化算法，提升程序处理速度和读数准确率；
 - 6. 选用更优的灰度化处理算法，让表盘和与背景区分度提高；
 - 7. 增强程序的智能性，适当调整代码，让程序**自动识别场景与背景**，自适应调节；
 - 8. 完善一些图像校正功能（图像增强修复，角度校正，锐化降噪处理等）；
 - 9. 希望经过处理，能读入连续的动图，因为汽车的速度变化是连续的过程，只有**能读取动图数据并能分析动图数据**，才能部署在车载系统上，才有商业价值。

这是从论文上摘抄下来的仪表盘识别的相对误差表，也是本系统进一步开发所应追求到的目标：



表 5-3-2 论文引用 相对误差表

序号	分类	人工读取	本文方法读取	相对误差/%
1	正常	7.10	7.078	0.22%
2		7.20	7.128	0.72%
3		5.10	5.084	0.16%
4		5.80	5.67	1.30%
5	遮挡	7.10	7.082	0.18%
6		7.20	7.138	0.62%
7	角度偏移	5.60	5.774	1.74%
8		7.20	7.134	0.66%
9	光照	3.60	3.612	0.12%
10		5.70	5.704	0.04%

五、收获与体会

历经多天多个难以言喻的白天与夜晚，总算将本作品给实现出来了，期间遇到了很多困难。每个环节都有每个环节的难处，但是每次当问题一步一步看到曙光之后，就十分开心和欣慰，恍然大明白的感觉。

当然到后面发现这也是一个进步的过程，可以从中发现更多问题，然后进一步优化处理，完成这个程序，也发现原来独立完成一份作品，是一件多么不容易的事情。

总的来说，整个流程下来还是收获满满，特别是学习到了很多知识，当然对 matlab 这个软件也更加熟练。除此之外，还巩固了课堂上的知识点，也让自己对这一门课有了更高的兴趣与学习热情。



第六章 致谢

经过一学期的学习和大作业项目的实际操作，我对《数字图像处理》这门课程有了更深刻的理解和体会。数字图像处理涉及数学、光学、电子学、计算机科学、计算机图形学、人工智能、模式识别、神经网络、模糊理论以及摄影等众多学科领域，其理论和技术体系十分庞大和复杂，包括“数字”“图像”和“处理”三个层面。

“图像”层面上，“图”是物体反射或者透射电磁波的分布，“像”是人的视觉系统对接收的图信息在大脑中形成的印象或认识。图像即是“图”和“像”两者的结合，是用各种观测系统以不同形式和手段观测客观世界而获得的、可以直接或间接作用于人的视觉系统而产生的视知觉实体。

“数字”层面上，是指图像的本质信息是数字，0-1 二进制的本质也是数字，对模拟图像进行数字化，可以将模拟图像转换成数字图像。

“处理”层面上，是指对数字图像信息进行加工，以改善图像的视觉效果和提高图像的实用性，或对数字图像进行压缩编码以减少所需的存储空间的技术。故，该处理也称为计算机图像处理，泛指利用计算机技术对数字图像进行某些数学运算和各种加工处理。例如，通过线性平滑滤波消除图像中的噪声、通过对比度拉伸或空间锐化滤波进行图像增强等，在像素级上进行的处理都属于处理的范畴。

可见，数字图像处理的基本特征是图像处理系统的输入和输出都是图像，也即是一个从图像到图像的过程。其研究内容和方法包图像增强、图像恢复、图像压缩编码、图像分割、图像特征提取、图像表示和描述、图像变换、图像的基本运算、频率域图像处理、小波图像处理、彩色图像处理、形态学图像处理、视频图像处理等。可见，本项目达到了课程的教学目标。

在作品开发过程中，除了受到了网上百度各方强者文章的帮助外，期间还得到了不少同学的帮助与启发。同时，感谢 20 级覃梓鑫同学的 1r580 系列表情包，为本系统设定了一个合适的 logo 图片，且在本实验文档中多处也放置了该系列的表情包。

本次实验项目的开发过程，使用了个人 Github 账号（主页：<https://github.com/starsinhands>）进行代码托管，最终成品的完整项目代码也同步上传到了个人的 Github 账号平台：

<https://github.com/starsinhands/Cars-info-Recognition-System>

最后还要感谢梁艳老师一学期以来的帮助与指导，让我对《数字图像处理》这一门课或者说是技术有了很大的兴趣与学习热情。学习这些选修课程，要学出真本事，不仅需要培养兴趣，更要在抽象问题上常作构思，常作想象，需要付出时间和精力。在未来的课程学习之路上，我也相信，数字图像处理课程给我带来的知识和阅历足以让我走得更深更远。



第七章 参考文献

- [1] [指针式仪表的自动读数与识别 指针仪表识别-CSDN 博客](#)
- [2] [毕业设计-基于计算机视觉技术的指针式仪表智能识别系统-CSDN 博客](#)
- [3] Matlab 形态学图像处理：二值图像分割 标记连通区域和重心位置 删除连通区域—csdn 用户—AI 吃大瓜
- [4] [MATLAB 指针式仪表自动读数系统设计-CSDN 博客](#)
- [5] [双指针仪表盘读数\(二\) ——表盘位置识别 指针仪表 多圈刻度圈识别-CSDN 博客](#)
- [6] [与霍夫变换有关的 Matlab 函数 :hough;houghpeaks;houghlines matlab houghpeaks-CSDN 博客](#)
- [7] [Matlab 中 bwmorph 函数的使用 bwmorph 函数 matlab-CSDN 博客](#)
- [8] 数字图像处理（MATLAB 版）——冈萨雷斯 中文高清版
- [9] [imopen 和 bwmorph 形态学笔记-CSDN 博客](#)
- [10] [图像处理函数 im2bw 详解-CSDN 博客](#)
- [11] blog.csdn.net/cxf7394373/article/details/5526034
- [12] 数字图像处理（第 4 版）——李俊山 清华大学出版社
- [13] [标准 HoughLines\(\) 变换-CSDN 博客](#)
- [14] [Python cv.HoughLines\(\) 方法参数与用法详解-CSDN 博客](#)
- [15] [Task05. 阈值分割——二值化 inrange 自动阈值二值化-CSDN 博客](#)
- [16] [图像处理函数——colfilt、bwperim、bwlabel、bwareaopen-CSDN 博客](#)
- [17] 黄杨乐天, 刘宜胜, 陈铤等. 基于深度学习的指针仪表识别系统[J]. 自动化与仪表, 2023, 38(11):100-104+108. DOI:10.19557/j.cnki.1001-9944.2023.11.021
- [18] 徐红武, 赵海月, 樊玉平等. 基于混合策略的指针式仪表刻度识别方法[J]. 机电工程技术, 2023, 52(10):184-188.
- [19] 吴家飞. 指针式仪表智能识别技术研究[J]. 装备制造技术, 2023, (10):186-189.

附录

一、图像素材库

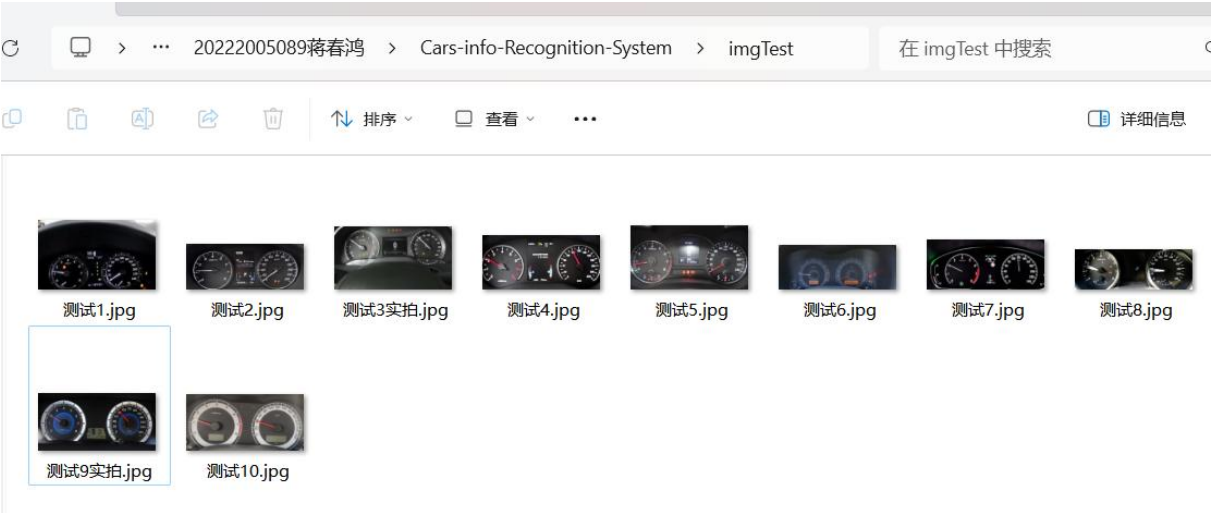


图 8-1-2 图像素材库

二、代码架构文件包



图 8-2-1 实验代码架构文件包

三、程序运行操作步骤



图 8-3-1 步骤 1 打开 matlab

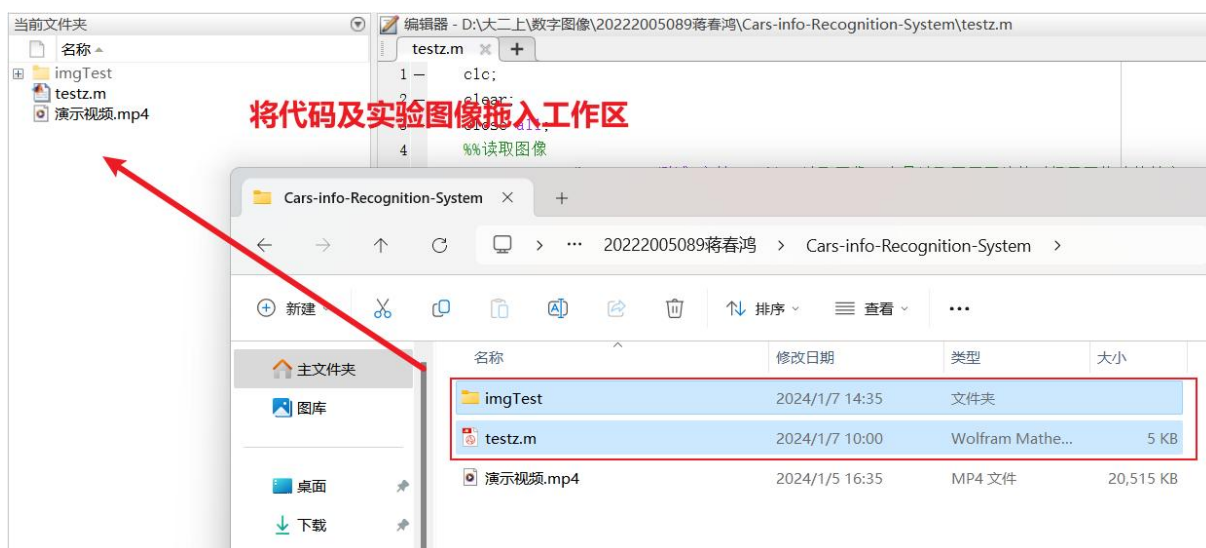


图 8-3-2 步骤 2 拖入素材与程序代码

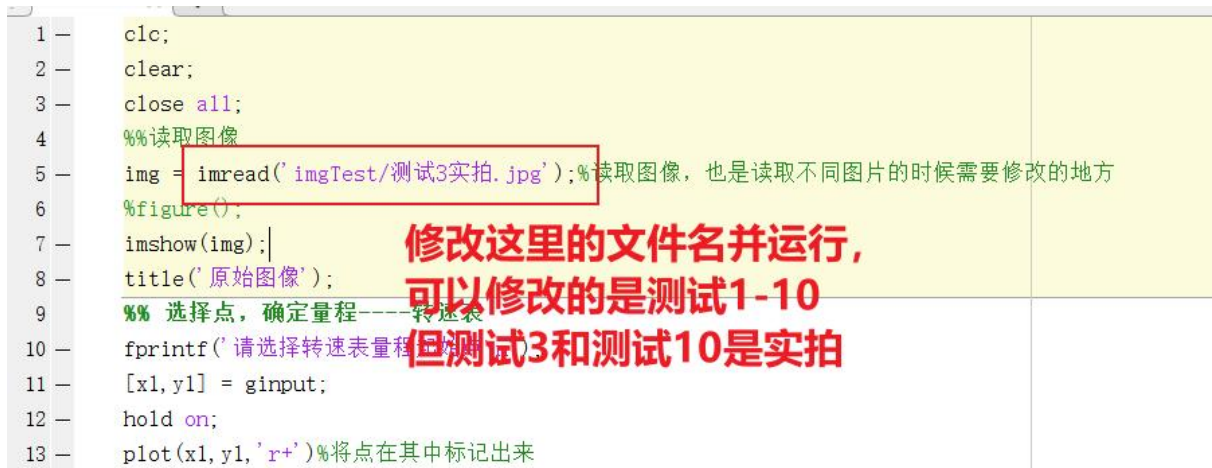


图 8-3-3 步骤 3 输入语句

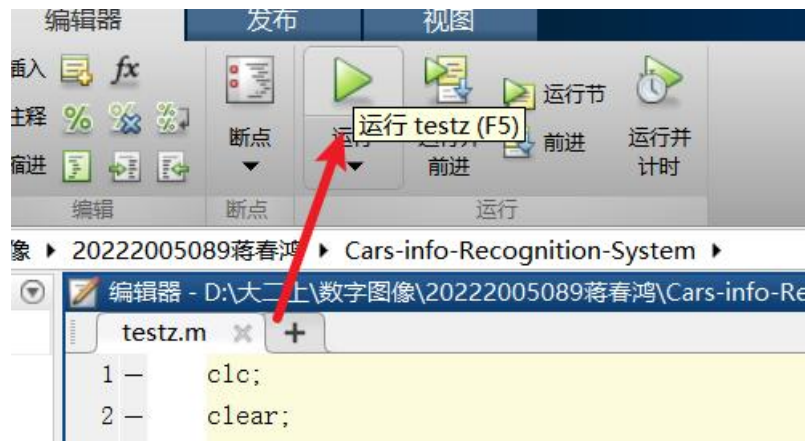


图 8-3-4 步骤 4 点击上方的运行

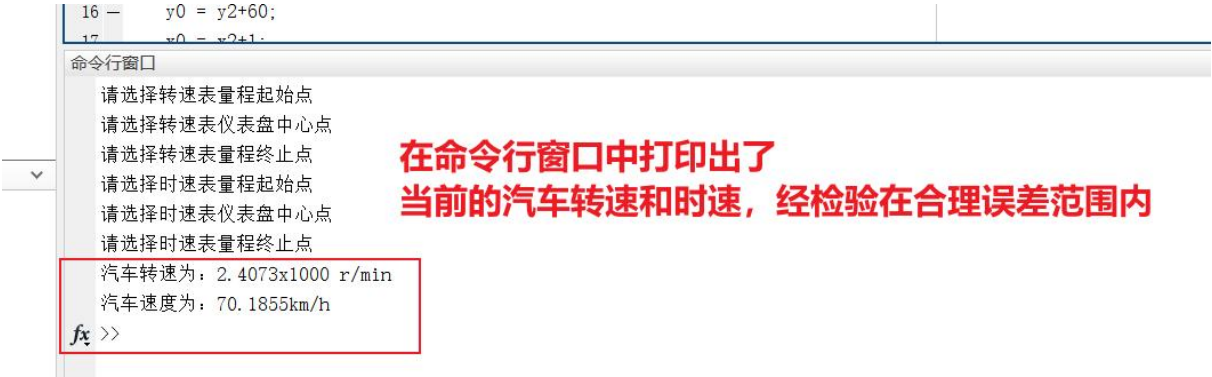


图 8-3-5 步骤 5 依次选点



软件依次运行产生 Figure 1-7 窗口





组图 8-3-6 步骤 6 运行代码，显示结果

