

影院票务辅助系统中座位信息的自动提取

刘思樊, 杨 洋, 童恒建, 唐杰辰

(中国地质大学(武汉) 计算机学院, 湖北 武汉 430000)

摘要:在电影院的票务系统中,一般分为线上售票和线下售票两种方式。目前线上售票有成熟的软件工具,用户可以通过手机或电脑在线上自动选择各种类型的座位信息。但在线下售票中,操作人员只能通过人工观察票务系统软件的屏幕界面,人工获取某个场次的不同类型的空余座位信息,然后再手工输入空余的座位信息到出票系统,打印纸质电影票后,然后进行售票。针对线下售票需要人工操作这种不方便不快捷的情况,作者利用 C++ 语言,基于 OpenCV 图像处理库,首先快速读取票务系统软件的界面图片的彩色图像,将三通道的彩色图像转化为单通道的灰度图像。利用 findContours() 函数获取图像的轮廓集,处理掉缺失和过小的轮廓后,通过外接矩形来获取轮廓的位置,将轮廓和外接矩形作为属性初始化对应的每个座位的实例化对象。通过读取各个对象的矩形内的像素点信息,可以确定对象的颜色以及位置。实现了对屏幕中各种不同类型的座位信息的自动识别与定位,准确率为 100%;然后,自动输入到出票系统,自动打印出纸质电影票,达到了自动线下售票的目的。开发此系统使用的技术方法,对于需要识别第三方软件界面的屏幕信息及其变化的场景,如:监控系统屏幕的变化,有一定的借鉴意义。

关键词:OpenCV;轮廓提取;图像处理;二值化;数学形态学

中图分类号:TP311.1;TP399

文献标识码:A

文章编号:1673-629X(2024)04-0193-05

doi:10.20165/j.cnki.ISSN1673-629X.2024.0029

Automatic Extraction of Seat Information in Cinema Ticketing Assistance System

LIU Si-fan, YANG Yang, TONG Heng-jian, TANG Jie-chen

(School of Computer Science, China University of Geosciences, Wuhan 430000, China)

Abstract: In the ticketing system of movie theaters, it is generally divided into two ways: online ticket sales and offline ticket sales. At present, there are mature software tools for online ticket sales, and users can automatically select various types of seat information online through mobile phones or computers. However, in offline ticket sales, the operator can only manually obtain different types of vacant seat information for a certain session by manually observing the screen interface of the ticketing system software, and then manually enter the vacant seat information into the ticketing system, print paper movie tickets, and then sell tickets. In view of the inconvenient and unfast situation that offline ticket sales require manual operation, the author uses C++ language, based on OpenCV image processing library, to first quickly read the color image of the interface picture of the ticketing system software, and convert the three-channel color image into a single-channel grayscale image. The findContours() function is used to obtain the contour set of the image, and after removing the missing and too small contours, the contour position is obtained through the external rectangle, and the contour and the external rectangle are initialized as the corresponding instantiated objects of each seat. By reading the pixel information within the rectangle of each object, you can determine the color and position of the object. It realizes the automatic identification and positioning of various types of seat information on the screen, with an accuracy rate of 100%. Then, it is automatically entered into the ticketing system and automatically prints out the paper movie ticket, which achieves the purpose of automatic offline ticket sales. The technical methods used in the development of this system have certain reference significance for scenarios that need to identify the screen information of third-party software interfaces and their changes, such as monitoring the changes of the system screen.

Key words: OpenCV; contour extraction; image processing; binarization; mathematical morphology

收稿日期:2023-05-26

修回日期:2023-09-27

基金项目:国家自然科学基金资助项目(41171339, U1803117)

作者简介:刘思樊(2000-),男,硕士,研究方向为遥感图像处理与分析;通信作者:杨 洋(2000-),女,硕士,研究方向为遥感图像处理与分析;童恒建(1970-),男,教授,博士,研究方向为遥感图像处理与分析。

0 引言

在电影院的票务系统中,一般有基于浏览器/服务器模式的线上售票系统,客户可以通过手机和电脑实现线上购票;然而,仍然有部分客户直接到电影院窗口进行现场购票,对于这种线下购票情况,售票员要通过人工观察票务系统软件的界面图像,找到空余的座位信息,然后再手工输入空余的座位信息到出票系统,打印纸质电影票后,再进行售票。这种人工观察界面图像,人工记录和人工输入空余的座位信息等操作,花费的时间长,当很多人在窗口排队买票时,而电影马上就要放映时,就易产生矛盾。该文的研究就是针对这种情况进行开展的,目的是自动提取票务系统软件的界面图像上的空余座位信息,达到自动售票的要求。该文基于 OpenCV 图像处理算法库,实现了自动读取图片的所有的座位信息,自动识别出每个座位的颜色信息并进行定位,对不同类型的座位信息进行分类汇总,准确率 100%,达到了自动线下售票的目的。当用户使用的第三方软件没有提供二次开发库时,而用户需要提取软件界面的信息或者检测所用的软件界面的变化信息时,如:安全监控软件界面中任一位置的颜色发生变化,该文所用的方法和技术对这类应用场景具有重要的借鉴意义。

1 图像处理

OpenCV 中提供了许多针对图像进行处理的函数,而要对图像进行轮廓提取,需要先对图像进行二值化处理,形态学降噪。具体处理流程如图 1 所示。

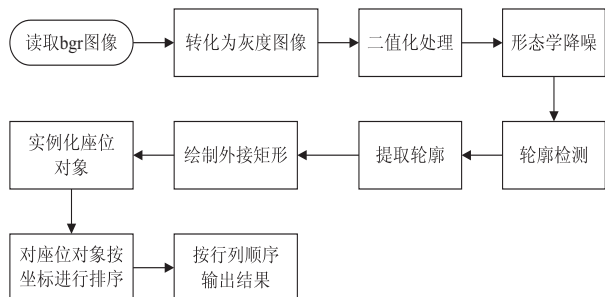


图 1 处理流程

1.1 二值化处理

在将三通道的彩色图像转化为单通道的灰度图像后,就可以对图像进行二值化处理。所谓的二值化处理,就是指转化掉图像中所有高于阈值或者低于阈值的像素点,若图像的阈值点设置为 T ,则所有像素值大于 T 的像素点赋值为 255,所有像素值低于 T 的像素点赋值为 0。 $f(x,y)$ 表示原图像在 (x,y) 点的像素值^[1], $g(x,y)$ 表示处理后图像在 (x,y) 点的像素值^[2]。

$$g(x,y) = \begin{cases} 255 & f(x,y) > T \\ 0 & f(x,y) \leq T \end{cases} \quad (1)$$

当 T 是一个常数时,上面公式的处理方式为全局阈值处理^[3],当 T 是一个变量时,上面公式的处理方式为可变阈值处理。当目标像素与背景像素的差异较大时,适用于对整个图像使用全局阈值处理^[4]。

初始图像、灰度图像、二值图像分别如图 2~4 所示。



图 2 初始图像



图 3 灰度图像

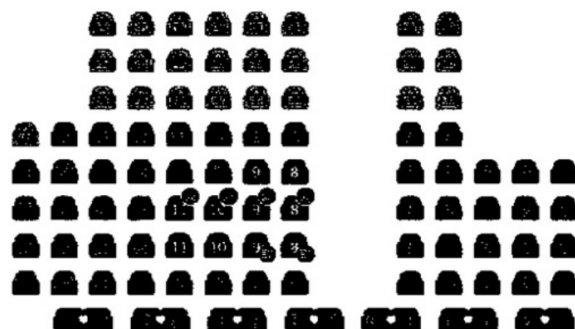


图 4 二值图像

1.2 使用 Otsu 算法的全局阈值处理

Otsu 算法能够完全基于图像的直方图进行计算,计算简单,不受对比度的影响,能够最大化图像前景与背景之间的方差^[5],使得图像分割时错分的概率最小。这种方法的基本思路就是给出不同类别中最优分离的一个阈值^[6],即使得类间方差最大化。对于 $M \times N$ 个像素的图像,将灰度值分为 $\{0,1,\dots,L-1\}$ 等 L 个灰度值。令 n_i 表示灰度值为 i 的像素个数。即 $MN = n_0 + n_1 + \dots + n_{L-1}$,归一化处理有 $p_i = n_i/MN$,即:

$$\sum_{i=0}^{L-1} p_i = 1, p_i \geq 0 \quad (2)$$

图像阈值 $T(k) = k, 0 < k < L-1$, 阈值将图像像

素点分为两类: c_1, c_2 , $c_1 = [0, k]$, $c_2 = [k + 1, L - 1]$ 。
图像属于 c_1 的概率为 $P_1(k)$ [7]。

$$P_1(k) = \sum_{i=0}^k P_i \quad (3)$$

c_1, c_2 中像素的平均灰度值为:

$$m_1(k) = \sum_{i=0}^k iP(i/c_1) = \sum_{i=0}^k iP(c_1/i)$$

$$P(i)/P(c_1) = \frac{1}{p_1(k)} \sum_{i=0}^k ip_i$$

$$m_2(k) = \frac{1}{p_2(k)} \sum_{i=k+1}^{L-1} ip_i \quad (4)$$

整个图像的平均灰度值为:

$$m_G = \sum_{i=0}^{L-1} ip_i \quad (5)$$

由式3~5可以计算类间方差:

$$\sigma^2(k) = P_1(k) (m_1(k) - m_G)^2 + P_2(k) (m_2(k) - m_G)^2 \quad (6)$$

只需要将所有满足 k 的整数代入公式计算, 得出 $\sigma^2(k)$ 的最大值, 对应的 k 值即是 Otsu 算法得出的对于图像的最佳阈值。在图像处理领域, 由于二值化处理能够极大地压缩数据量, 简化对图像进行处理的步骤并且能够凸显图片的轮廓, 所以二值化处理成为了许多图像处理领域中必要的预处理过程。

1.3 形态学图像处理

在图像的形态学处理中, 腐蚀和膨胀是两个最基本的运算, 也是其他许多形态学运算的基础[8]。而在二值图像中, 图像的集合是二维整数空间的成员。在图像处理时, 使用目标元素和结构元素 (Structure Element, SE) 两个像素集合, 腐蚀和膨胀就是基于这两个像素集合进行的操作。定义目标元素集合为 A , 结构元素集合为 B , 集合 B 相对于原点的反射为 \hat{B} , 具体定义为:

$$\hat{B} = \{w \mid w = -b, b \in B\} \quad (7)$$

可知, 对于集合 B 中的坐标 (x, y) , 相应的 \hat{B} 中的坐标为 $(-x, -y)$ 。集合 $(B)_z$ 是集合 B 相对于点 $z = (z_1, z_2)$ 的平移, 具体定义为[9]:

$$(B)_z = \{c \mid c = b + z, b \in B\} \quad (8)$$

可知, 对于集合 B 中的坐标 (x, y) , 相应的 $(B)_z$ 中的坐标为 $(x + z_1, y + z_2)$ [10]。在形态学处理中, 对于集合 B 对集合 A 的腐蚀 (过程如图5所示) 表示为 $A \ominus B$, 具体定义为[11]:

$$A \ominus B = \{z \mid (B)_z \subseteq A\} \quad (9)$$

对于集合 B 对集合 A 的膨胀 (过程如图6所示) 表示为 $A \oplus B$, 具体定义为:

$$A \oplus B = \{z \mid (\hat{B})_z \cap A \neq \emptyset\} \quad (10)$$

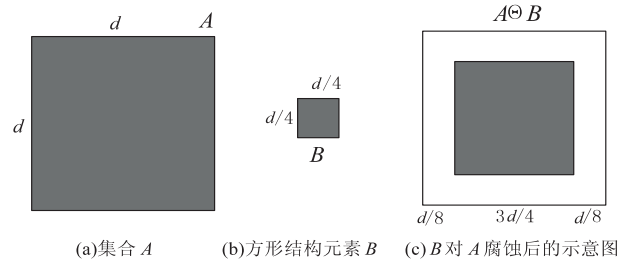


图5 腐蚀过程

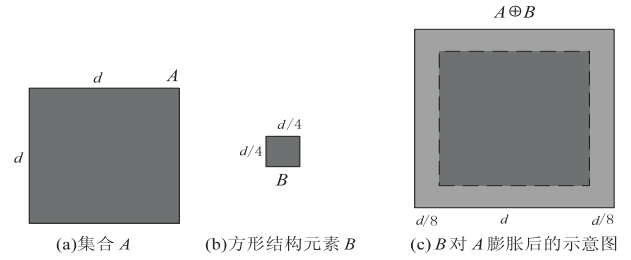
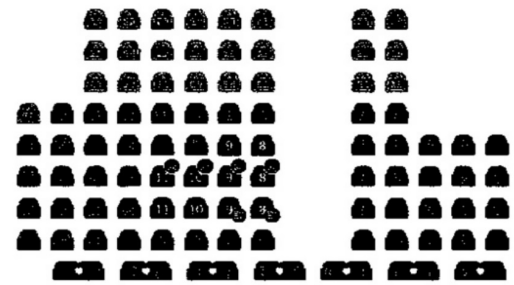
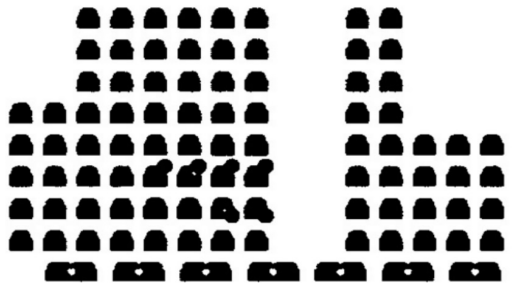


图6 膨胀过程

形态学处理中的开闭运算就是对腐蚀和膨胀操作的组合应用[12], 其中集合 B 对集合 A 的开运算就是首先 B 对 A 腐蚀, 然后再进行膨胀操作, 开运算一般会平滑物体的轮廓、断开较窄的狭颈并消除较细的突出物[13]; 集合 B 对集合 A 的闭运算就是首先 B 对 A 膨胀, 然后再进行腐蚀操作, 闭运算同样也会平滑轮廓的一部分, 但与开运算相反, 它通常会弥合较窄的间断和细长的沟壑, 消除较小的孔洞, 填补轮廓线中的断裂。在本次应用中, 使用开运算减少了图像中一些小的噪点, 实现了降噪功能。形态学降噪如图7所示。



(a) 二值化图像



(b) 形态学降噪后的二值化图像

图7 形态学降噪

2 轮廓提取

2.1 轮廓检测

图像轮廓提取的一个重要基础就是图像的边缘检

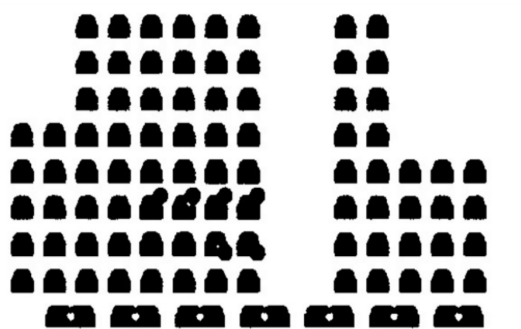
测,目前图像的边缘检测是计算机图像处理研究领域的热门问题,也是许多图像分析问题的重要基础^[14]。目前基于 OpenCV 的轮廓检测算子主要有 Sobel 算子和 Canny 算子,Sobel 算子的优势就是对于图像要求不高,对于图像的噪声处理比较优秀,而缺点就是轮廓边缘的定位精度不高;Canny 算子则对于轮廓边缘的定位精度更高,但也可能会存在图像的噪声干扰检测结果的情况^[15]。

在完成图像二值化和降噪操作以后,就可以使用 OpenCV 中的 `findContours(thresh, contours, hierarchy, cv2::RETR_TREE, cv2::CHAIN_APPROX_SIMPLE)` 函数进行轮廓提取。函数中的第一个参数是输入的二值图像;第二个参数为输出的轮廓集合;第三个参数为输出的每条轮廓的属性,表示后一个轮廓,前一个轮

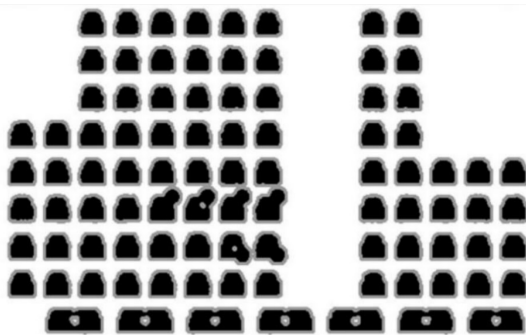
廓,父轮廓,内嵌轮廓的索引编号;第四个参数表示用一个等级树的结构来构建轮廓信息,外层轮廓可以包含内嵌轮廓;第五个参数表示仅保存轮廓的拐点信息,放弃轮廓两个拐点之间直线的信息点。

2.2 保存轮廓数组并绘制轮廓

在使用 `findContours()` 函数读取轮廓时,可以使用一个二维的点向量组来保存轮廓数据,具体为 `vector<vector<Point>>` 类型。可以通过向量对象,快速遍历每条轮廓,使用 `contourArea()` 可以获取轮廓的面积,通过一次遍历,可以删除掉面积过小的轮廓以及内轮廓。在轮廓数组的双重点向量中,每个点向量表示一条轮廓,在筛选出异常轮廓后,可以通过 `drawContours()` 函数在原图像中绘制轮廓信息,如图 8 所示。



(a)二值化图像



(b)图像轮廓示意图

图 8 绘制轮廓

3 确定单位信息

通过 `boundingRect()` 函数绘制轮廓的外接矩形,通过外接矩形获取原图像中某个矩形框内部的像素点的值,可以通过遍历某个区域的所有像素点,来确定相应座位模块的颜色,用外接矩形以及颜色可以实例化相应的座位模块。

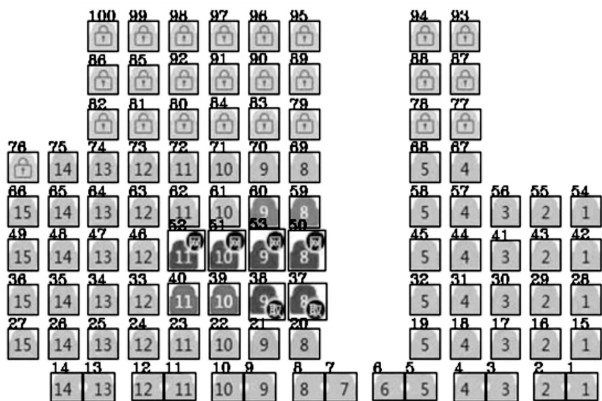


图 9 绘制外接矩形

具体的数据结构如下:

```
struct SeatColorInformation
{
```

//构造函数,用外接矩形,面积,颜色和编号来构造座位对象

```
Seat(Rect rect, float area, int color_index, int count) {
    this->rect = rect;
    this->area = area;
    this->color_index = color_index;
    this->count = count;
}

int count; //座位的编号
float area; //座位的面积
Rect rect; //当前座位的外接矩形
int color_index; //座位的颜色信息
int row; //座位的行
int col; //座位的列
bool flag = 0;
};
```

在实例化所有的座位模块以后,可以通过外接矩形的函数 `rect.tl().x`,返回矩形左上点坐标的 x 值;通过函数 `rect.tl().y`,返回矩形左上点坐标的 y 值,根据 `sort()` 函数可以按照相应 x, y 值进行排序,确定座位模块的行、列位置。在确定所有信息以后,可以通过程序输出所有颜色相应的座位的行列位置,如图 10 所示。


```

座位个数: 100 总面积为: 55451
包含双人座位: 7个

座位颜色: 蓝色 数量: 63
[第1行 第1列 ] [第1行 第2列 ] [第1行 第3列 ] [第1行 第4列 ] [第1行 第5列 ]
[第1行 第6列 ] [第1行 第7列 ] [第1行 第8列 ] [第1行 第9列 ] [第1行 第10列 ]
[第1行 第11列 ] [第1行 第12列 ] [第1行 第13列 ] [第1行 第14列 ] [第2行 第1列 ]
[第2行 第2列 ] [第2行 第3列 ] [第2行 第4列 ] [第2行 第5列 ] [第2行 第10列 ]
[第2行 第11列 ] [第2行 第12列 ] [第2行 第13列 ] [第2行 第14列 ] [第2行 第15列 ]
[第3行 第1列 ] [第3行 第2列 ] [第3行 第3列 ] [第3行 第4列 ] [第3行 第5列 ]
[第3行 第12列 ] [第3行 第13列 ] [第3行 第14列 ] [第3行 第15列 ] [第4行 第3列 ]
[第4行 第1列 ] [第4行 第2列 ] [第4行 第4列 ] [第4行 第5列 ] [第4行 第12列 ]
[第4行 第13列 ] [第4行 第14列 ] [第4行 第15列 ] [第5行 第1列 ] [第5行 第2列 ]
[第5行 第3列 ] [第5行 第4列 ] [第5行 第5列 ] [第5行 第10列 ] [第5行 第11列 ]
[第5行 第12列 ] [第5行 第13列 ] [第5行 第14列 ] [第5行 第15列 ] [第6行 第4列 ]
[第6行 第5列 ] [第6行 第8列 ] [第6行 第9列 ] [第6行 第10列 ] [第6行 第11列 ]
[第6行 第12列 ] [第6行 第13列 ] [第6行 第14列 ]

座位颜色: 橙色 数量: 2
[第5行 第8列 ] [第5行 第9列 ]

座位颜色: 红色 数量: 6
[第3行 第8列 ] [第3行 第9列 ] [第4行 第8列 ] [第4行 第10列 ] [第4行 第11列 ]
[第4行 第9列 ]

座位颜色: 粉色 数量: 2
[第2行 第8列 ] [第2行 第9列 ]

座位颜色: 灰色 数量: 25
[第6行 第15列 ] [第7行 第4列 ] [第7行 第5列 ] [第7行 第8列 ] [第7行 第11列 ]
[第7行 第12列 ] [第7行 第13列 ] [第7行 第9列 ] [第7行 第10列 ] [第8行 第12列 ]
[第8行 第13列 ] [第8行 第4列 ] [第8行 第5列 ] [第8行 第8列 ] [第8行 第9列 ]
[第8行 第10列 ] [第8行 第11列 ] [第9行 第4列 ] [第9行 第5列 ] [第9行 第8列 ]
[第9行 第9列 ] [第9行 第10列 ] [第9行 第11列 ] [第9行 第12列 ] [第9行 第13列 ]

座位颜色: 绿色 数量: 2
[第3行 第10列 ] [第3行 第11列 ] [ INFO:0 ] global C:\build\master_winpack-build-win64-vc14\opencv\modules\highg

```

图10 输出结果

4 结束语

最初笔者使用的技术方案是应用第三方的数字识别软件识别图像中的数字字符。原来认为所有数字都是印刷体,而且很清晰,应该能100%识别;然而通过实验证明,并不能达到100%的识别率。但是,甲方要求必须100%识别,得出准确的剩余座位信息。如果不是100%识别,出票系统就会出现错误,造成座位信息冲突,引发矛盾。笔者测试了多家数字识别软件,包括开源的和大型公司的产品,都不能达到100%的准确率。后来,通过分析,每一个座位的位置信息在图像中的相对位置是固定的,不同的颜色表示不同的销售情况;通过上文所述的方法,获取每个座位的外接矩形,就获得了每个座位的位置信息,通过获取每个矩形内的颜色,就知道了每个位置的销售信息,从而获得了100%的准确率。该方法简单快捷,具有更高的实用性。类似的场景,只要是显示屏幕中待识别的对象位置信息相对固定,而且用颜色就能区分的,就可以用提供的技术方案。如:彩色的LED灯的管理,通过程序识别出不同颜色的LED灯的位置和状态,能够方便进行开关管理以及亮度调整等。

参考文献:

- [1] 张艺,匡毅,王梅,等.基于OpenCV的人体轮廓检测算法[J].计算机技术与发展,2020,30(8):81-85.
- [2] MITIANOUDIS N,PAPAMARKOS N. Document image binarization using local features and Gaussian mixture modeling[J]. Image and Vision Computing,2015,38(3):33-51.
- [3] 王林,董楠.基于Gabor特征与卷积神经网络的人体轮廓提取[J].南京理工大学学报:自然科学版,2018,42(1):89-95.
- [4] 张青苗,李鹏,赵勇,等.一种基于OpenCV实现的运动目标检测算法的研究[J].舰船电子工程,2010,30(7):59-61.
- [5] 马娅丽,熊淑华,黑建业.一种改进的LOG边缘算法研究[J].计算机技术与发展,2013,23(9):6-9.
- [6] HUANG Y,LIU M,LIU X,et al. Research on recognition method of neutron-gamma single pulse pileup based on mathematical morphology[C]//International conference on intelligent computing and signal processing. Xi'an: IEEE, 2021:586-589.
- [7] 张启轩,袁明辉.基于OpenCV的物体图像边缘缺陷识别研究[J].软件导刊,2021,20(4):231-235.
- [8] 吴凤和.基于计算机视觉测量技术的图像轮廓提取方法研究[J].计量学报,2007,28(1):18-22.
- [9] GONG X,SU H,XU D,et al. An overview of contour detection approaches[J]. International Journal of Automation and Computing,2018,15(6):656-672.
- [10] 韦德鹏,陈继清,罗天,等.基于改进八方向Sobel算子的图像轮廓提取方法[J].现代电子技术,2022,45(19):54-58.
- [11] 邹柏贤,林京壤.图像轮廓提取方法研究[J].计算机工程与应用,2008,44(25):161-165.
- [12] 徐国雄,张骁,胡进贤,等.基于阈值分割和轮廓提取的图像边缘检测算法[J].计算机技术与发展,2015,25(12):64-67.
- [13] 李牧子.基于图像识别的目标检测与跟踪系统的设计与实现[D].北京:北京邮电大学,2019.
- [14] 高薪,胡月,杜威,等.腐蚀膨胀算法对灰度图像去噪的应用[J].北京印刷学院学报,2014,22(4):63-65.
- [15] ZHOU W,FEI Z,HU H,et al. Real-time object subspace searching based on discrete searching paths and local energy[J]. International Journal of Automation and Computing, 2016,13(2):99-107.