

doi: 10.12052/gdutxb.210038

高分辨率桥梁裂缝图像实时检测

刘信宏, 苏成悦, 陈静, 徐胜, 罗文骏, 李艺洪, 刘拔

(广东工业大学 物理与光电工程学院, 广东 广州 510006)

摘要: 针对现有桥梁裂缝检测算法实时性弱、可靠性差等问题, 提出一种嵌入式平台上的实时检测算法。使用移动平均法粗分割, 依据几何特征和区域生长法再分割, 定位候选裂缝片段; 基于裂缝先验条件, 建立双判别准则的裂缝聚合模型, 递归合并裂缝片段并抑制干扰。实验表明, 算法能有效提取细小裂缝, 抑制不均匀光照和污渍等复杂背景干扰, 识别性能与数种现有算法相比提高115%以上; 在嵌入式开发板上处理1500万像素的图像仅耗时1.73 s。

关键词: 图像处理; 桥梁裂缝; 高分辨率; 实时检测; 嵌入式平台

中图分类号: TP391

文献标志码: A

文章编号: 1007-7162(2022)06-0073-07

Real Time Detection of High Resolution Bridge Crack Image

Liu Xin-hong, Su Cheng-yue, Chen Jing, Xu Sheng, Luo Wen-jun, Li Yi-hong, Liu Ba

(School of Physics and Optoelectronic Engineering, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: The problem of current algorithms is lack of timeliness and reliability in bridge crack detection. In this paper, a real-time algorithm on embedded platform is proposed. Firstly, moving average method is used to segment the image coarsely. Then, candidate crack fragments are selected by region growing method using the geometric features of their contours. Finally, a crack merging model with two criteria is built to merge the crack fragments recursively and suppress interference, based on the prior condition of bridge cracks. Experimental results show that the proposed method performs better than several existing methods by 115% at least, especially on extracting hairline cracks and complex cases with uneven illumination and dirty mark. Dealing with an image with 15 megapixel on embedded platform, it costs only 1.73 s.

Key words: image processing; bridge crack; high resolution; real-time detection; embedded platform

混凝土结构桥梁的裂缝会破坏桥梁结构整体性^[1], 甚至引发坍塌事故, 定期巡检桥梁具有必要性。国内外学者对桥梁裂缝检测做了大量研究, 如k-means结合区域生长^[2]、裂缝发展趋势估算的思想^[3]、基于局部纹理的检测算法^[4]、使用支持向量机、自适应提升(Adaboost)和随机森林的分类模型^[5]、照度补偿模型和裂缝概率图^[6]等。在统计信息网格聚类的思想下, 结合区域生长算法, 取得了较好的裂缝提取效果^[7], 但其网格聚类环节中, 容易选取到背景区域。应用多尺度Hessian矩阵可降低纹理干扰, 较好地抑制背景, 增强裂缝^[8]。但上述研究都没有考虑算法的实时性, 无法应用于嵌入式平台。多尺度的方向滤波

器, 降低了光照及背景干扰, 能在主机端进行较低分辨率实时检测^[9]。Canny边缘和区域生长的思想可实现快速裂缝分割^[10], 但抗干扰能力不足。还有运用U-Net模型^[11]、全连接网络^[12]和卷积神经网络^[13-15]等深度学习的方法, 跳过特征描述的过程, 实现端到端的识别, 其依赖于大量的数据集和图形处理器支持, 在移动平台较难实现在线实时检测。

本文提出一种嵌入式平台上高分辨率裂缝图像的实时检测算法。采用移动平均法^[16]粗分割; 通过几何特征筛选候选裂缝片段, 结合区域生长修复和过滤候选裂缝片段; 建立双判别准则的裂缝片段聚合模型, 递归合并裂缝片段, 并抑制干扰区域。

收稿日期: 2021-03-10

基金项目: 广东省科技计划项目(2017A020208063); 广州市科技计划项目(201804010384)

作者简介: 刘信宏(1995-), 男, 硕士研究生, 主要研究方向为图像处理、模式识别, E-mail: liuxinhong@mail2.gdut.edu.cn

通信作者: 苏成悦(1961-), 男, 教授, 博士, 主要研究方向为应用物理、图像识别, E-mail: cysu@gdut.edu.cn

1 裂缝检测算法

本文提出的裂缝实时检测算法流程如图1所示,图中二值图黑色为前景,白色为背景。

1.1 图像预处理

高斯滤波广泛运用于无人机图像处理、视频图像处理和图像预处理等场景^[17-18]。图像噪声中大部分噪声为白噪声,使用高斯滤波器可有效降低噪声对检测的影响^[19]。对输入图像进行高斯模糊,记 $I(x,y)$ 为高斯滤波后的灰度图像。

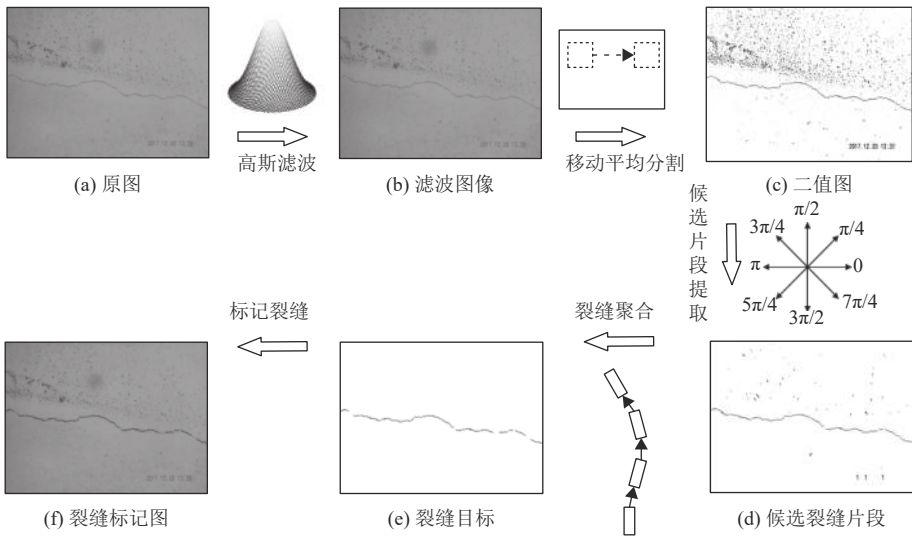


图1 裂缝识别流程图

Fig.1 Flow chart of the proposed method

1.2 候选裂缝片段提取

候选裂缝片段提取模块如图2所示。图2(a)为裂缝阈值分割图,通过第一次几何轮廓筛选得到图2(b)所示的种子区域,采用八方向区域生长算法获得图2(c),再进行一次几何轮廓筛选得到如图2(d)所示的候选裂缝片段,其连通域集合记为 $Q = \{Q_1, Q_2, \dots, Q_m\}$ 。

1.2.1 几何轮廓筛选

裂缝片段和背景噪声具有较明显的几何特征差异,如图3所示,图3(a)为背景噪声,其轮廓无明显几何规则;图3(b)为裂缝片段区域,具有明显的细长几何特征。

本文采用连通域面积 A 和连通域轮廓的惯性率 η_{IR} 来描述连通域的几何特征。二值图像的连通域面积为图像的零阶矩,图像的二维 $(i+j)$ 阶矩 M_{ij} 的定义如式(3)所示,则连通域面积 $A=M_{00}$ 。

$$M_{ij} = \sum_{(x,y) \in r_{x,y}} x^i y^j B(x,y) \quad (3)$$

式中: $i = 0, 1, 2, \dots, j = 0, 1, 2, \dots, r_{x,y}$ 为连通域内所包围

桥梁裂缝图像受复杂光照环境影响,其亮度分布不均匀。受Shi等^[16]的移动平均分析的启发,本文采取移动平均的分割方法,阈值由式(1)确定。

$$T(x,y) = \mu_I(x,y) + \alpha \quad (1)$$

式中: $\mu_I(x,y)$ 为图像 $I(x,y)$ 中尺寸为 $\varepsilon \times \varepsilon$ 的滑动窗口的灰度均值, α 为偏移系数。

获取二值图像 $B(x,y)$ 的分割过程如式(2)所示,前景目标区域设为1,背景区域设为0。

$$B(x,y) = \begin{cases} 1, & I(x,y) > T(x,y) \\ 0, & \text{others} \end{cases} \quad (2)$$

的点。

惯性率可用于描述目标的伸长程度,惯性率越接近1,则越相似于圆形。惯性率可由图像中心矩得

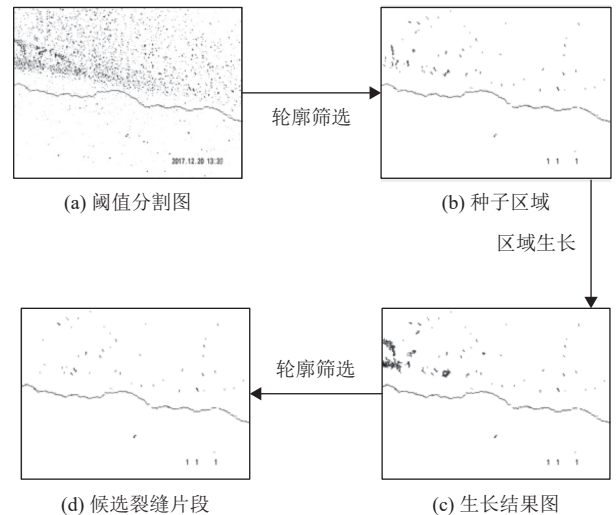


图2 候选裂缝片段提取流程

Fig.2 Flow chart of candidate crack fragment sampling

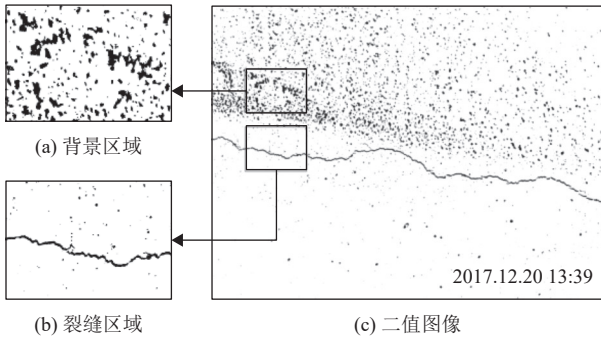


图3 裂缝与背景区域的几何特征差异

Fig.3 Difference between crack and back ground region

到,图像的二维 $(i+j)$ 阶中心矩定义如式(4)所示。

$$\omega_{ij} = \sum_{(x,y) \in R_{x,y}} (x-\hat{x})^i (y-\hat{y})^j B(x,y) \quad (4)$$

式中: $i=0,1,2,\dots, j=0,1,2,\dots$, 连通域的质心

$$(\hat{x}, \hat{y}) = \begin{pmatrix} M_{10} & M_{01} \\ M_{00} & M_{00} \end{pmatrix}.$$

惯性率定义如式(5)所示。

$$\eta_{IR} = \frac{\eta_a - \eta_b}{\eta_a + \eta_b} \quad (5)$$

式中: $\eta_a = \omega_{20} + \omega_{02}, \eta_b = \sqrt{4\omega_{11}^2 + (\omega_{20} - \omega_{02})^2}$ 。

为过滤背景干扰,定义轮廓筛选准则如式(6),

满足式(6)的连通域保留,否则被滤除。

$$A \in (T_{Amin}, T_{Amax}) \text{ 且 } \eta_{IR} < T_{IR} \quad (6)$$

式中: (T_{Amin}, T_{Amax}) 为轮廓面积阈值范围, T_{IR} 为轮廓惯性率阈值。

1.2.2 区域生长

区域生长是依据预设的生长规则,将图像像素或者子区域合并的过程。从一组种子点开始遍历相邻像素,寻找满足预设生长规则的像素,并将其添加到种子中而形成生长区域,常用于图像分割^[7,10]。

计算连通域的面积和惯性率,依据式(6)选择具有显著裂缝几何特征的连通域,将其轮廓像素作为区域生长的初始种子集合 $S = \{\{S_1\}, \{S_2\}, \dots, \{S_p\}\}$,以连通域的灰度均值作为全部种子灰度值 G_{sv} 。种子像素集合 S 在二值图像 $B(x,y)$ 中进行八邻域区域生长,步骤如下:

1) 从 S 中取出一个种子像素坐标 (x_s, y_s) , 将其从 S 中删除;

2) 遍历坐标 (x_s, y_s) 的八邻域像素 (x_{sr}, y_{sr}) , 若 $B(x_{sr}, y_{sr})$ 为0, 且灰度图 $I(x,y)$ 中对应像素满足式(7)所示的生长条件, 则将 $B(x_{sr}, y_{sr})$ 设为1, 并将坐标 (x_{sr}, y_{sr}) 加入 S 中;

$$|G_{sv} - I(x_{sr}, y_{sr})| < T_s \quad (7)$$

式中: T_s 为区域生长灰度阈值。

3) 重复步骤1)和2), 直到 S 为空集, 输出区域生长完成的二值图像 $B'(x,y)$ 。

1.3 裂缝聚合

如图4所示, 提取的候选裂缝不连续, 且与背景干扰的形态相似。本文提出双判别准则的裂缝聚合模型, 模拟裂缝延伸趋势, 依据位置和方向信息构造聚合准则, 递归聚合细长裂缝, 并抑制背景干扰。

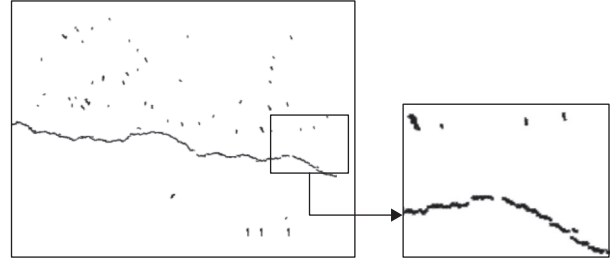


图4 裂缝片段

Fig.4 Crack fragment

1.3.1 聚合准则

以裂缝片段轮廓的最大距离点对 (P_b, P_f) 作为裂缝片段的端点, 联合裂缝片段最小外接矩形倾角 V , 构建裂缝片的特征向量 $H = (P_b, P_f, V)$, 其中, 本文通过轮廓的上下左右4个边界点的快速枚举来近似替代 P_b 和 P_f ; 使用Skalsky^[20]的方法检测裂缝片段的凸包后, 运用旋转卡壳法^[21]处理凸包而得到的矩形, 即为裂缝片段的最小外接矩形。记聚合裂缝集合为 $X = \{X_1, X_2, \dots, X_n\}$, 其中每个裂缝元素由多个裂缝片段构成。记裂缝片段最小外接矩形的长边为 l_w 。

定义两裂缝片段 Q_1 和 Q_2 的距离 $D(Q_1, Q_2)$ 为两裂缝片段的端点间的最短距离, 裂缝片段的聚合准则1定义如式(8)所示。

$$\begin{cases} D(Q_1, Q_2) < T_{dmin} \times \min(l_{w1}, l_{w2}) \\ |e_1 \cdot e_2| > \frac{1}{2} \end{cases} \quad (8)$$

式中: T_{dmin} 为短距离阈值, e_1 和 e_2 分别为 Q_1 和 Q_2 倾角方向的单位向量。

假设 Q_1 和 Q_2 间的最短距离点对为 (P_{f1}, P_{b2}) , 则有方向向量 $\vec{P_{f1}P_{b2}}, \vec{P_{f1}P_{b1}}$ 和 $\vec{P_{b2}P_{f2}}$, 定义裂缝片段的聚合准则2如式(9)所示。

$$\begin{cases} D(Q_1, Q_2) < T_{dmax} \times \min(l_{w1}, l_{w2}) \\ \vec{P_{f1}P_{b2}} \cdot \vec{P_{f1}P_{b1}} > 0 \text{ 且 } \vec{P_{f1}P_{b2}} \cdot \vec{P_{b2}P_{f2}} > 0 \end{cases} \quad (9)$$

式中: T_{dmax} 为长距离阈值。

定义裂缝过滤准则, 如式(10)所示。若 $\delta_k = 0$, 则将裂缝 X_k 剔除, 否则保留该裂缝。

$$\delta_k = \begin{cases} 1, & \sum_{i \in X_k} l_{wi} > T_{\text{CRL}} \\ 0, & \text{others} \end{cases}, k = 1, 2, \dots, n \quad (10)$$

式中: T_{CRL} 为裂缝长度阈值。

裂缝片段聚合算法流程如下:

- 1) 计算候选裂缝片段连通域集合 Q 中元素的特征向量 H ;
- 2) 按 $\max(P_{b,y}, P_{f,y})$ 降序和 $\min(P_{b,x}, P_{f,x})$ 升序对 Q 排序;
- 3) 对 Q 中满足式(8)或(9)的元素进行聚合, 输出聚合裂缝集合 X ;
- 4) 将集合 X 作为输入, 重复步骤3), 直到集合 X 和 Q 的元素个数相等;
- 5) 依据式(10)过滤裂缝。

其中步骤3)为算法的核心步骤, 其流程如表1所示。

1.3.2 裂缝聚合示例

裂缝片段聚合示例如图5所示, 图5(a)为初始裂缝片段; 图5(b)为第一次递归后的聚合结果, 产生4个子集, 分别为 $\{Q_1, Q_4, Q_6\}$ 、 $\{Q_2\}$ 、 $\{Q_3, Q_5, Q_7\}$ 和 $\{Q_8, Q_9\}$; 将每个子集作为一个裂缝片段, 重新初始化其指标向量 H , 如图5(c)所示; 图5(d)为第二次递归后的聚合结果, 将图5(b)中的 $\{Q_1, Q_4, Q_6\}$ 和 $\{Q_8, Q_9\}$ 聚

表1 裂缝聚合算法

Table 1 Algorithm of merging crack fragments

Input: Q : contours set of candidate crack fragments	
Output: X : cracks set	
1	$Q = \{Q_1, Q_2, \dots, Q_m\}, X = \emptyset$
2	for $i \in [1, m]$ do
3	select an unused Q_i and mark it
4	curr $Q = Q_i, X_i = \emptyset$
5	$X_i \leftarrow Q_i$ // Append Q_i to X_i
6	for $j \in [i+1, m]$ do
7	select an unused Q_j
8	//merge fragments using formula (8) and (9)
9	if ISMATCH(curr Q, Q_j) then
10	mark Q_j as the same as Q_i
11	$X_i \leftarrow Q_j$ // Append Q_j to X_i
12	curr $Q = Q_j$, and set the endpoints of curr Q as the farthest point pair of X_i
13	end if
14	end for
15	Regard X_i as a crack fragment and initialize its character vector H
16	$X \leftarrow X_i$ // Append X_i to X
17	end for

合为 $\{Q_{1_1} Q_{1_5}\}$ 。依据式(10)过滤裂缝, 最终得到 $\{Q_{1_1} Q_{1_5}\}$ 和 $\{Q_{3_1} Q_{3_3}\}$ 两条裂缝。

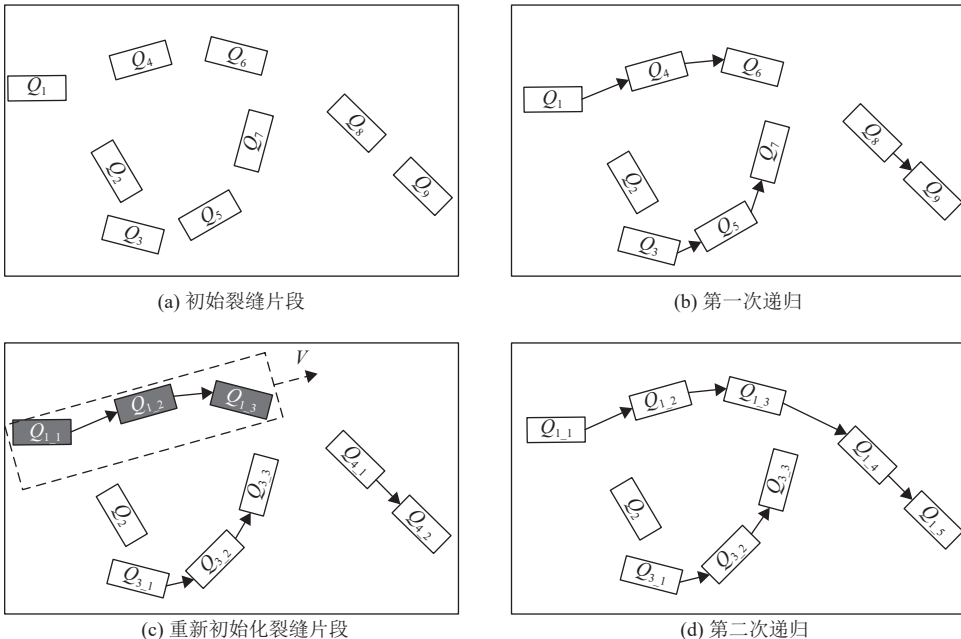


图5 裂缝片段递归聚合示例

Fig.5 Sample of merging crack fragments recursively

2 实验结果分析

本文选取若干代表性桥梁裂缝实拍图像进行测

试, 图像分辨率为 4608×3456 , 将本文算法与文献[7-8, 10]方法比较。测试平台为arm开发板: Raspberry Pi 4 Model B Rev, 处理器: ArmCortex-

A72@1.5 GHz, 内存:4GBLPDDR4 SDRAM, 操作系统:Linux raspberrypi 4.19.57-v7l+, 软件编程语言:C++。采用精确度(Precision)、召回率(Recall)和综合指标F值(F-measure) 3个指标进行评估,在图像分割中,其定义分别为 $\gamma_P = \frac{N_{gd}}{N_d}$, $\gamma_R = \frac{N_{gd}}{N_g}$, $\gamma_F = \frac{2\gamma_P\gamma_R}{\gamma_P + \gamma_R}$, 其中 N_{gd} 为正确分割的裂缝像素个数, N_d 为分割结果中的裂缝像素个数, N_g 为人工标注的裂缝像素个数。F值越大,表明检测性能越好。

2.1 灵敏性分析及参数确定

所研究裂缝图像的裂缝像素宽度范围为[2,14], 本文设滑动窗口尺寸 ε 为裂缝最大宽度的5倍,并取奇数值。针对所研究的图像数据集,统计发现单条裂缝的像素个数占图像总像素比例小于0.2%,同时为了排除小干扰区域,本文设轮廓面积阈值范围(T_{Amin} , T_{Amax})为 $(10^2, 3 \times 10^4)$ 。

本文算法待确定的核心参数有:轮廓惯性率阈值 T_{IR} 、区域生长灰度阈值 T_s 、短距离阈值 T_{dmin} 和长距离阈值 T_{dmax} 。设参数初始值为: $T_{IR} = 0.1$, $T_s = 10$, $T_{dmin} = 0.1$, $T_{dmax} = 0.8$, 分别独立改变参数值,对数据集全部图像进行测试。分别统计准确率、召回率和F值3个指标的均值,如图6所示。图6(a)表明, T_{IR} 在区间

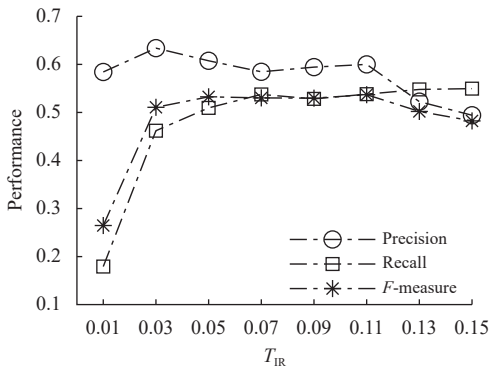
[0.05,0.12]范围内都有较好效果,选取F值开始下降的前一个数值0.11;图6(b)中, $T_s = 8$ 时,F值表现最佳;图6(c)和图6(d)中, T_{dmin} 和 T_{dmax} 的趋势一致,当参数值较小时,其精确度高,但召回率低,说明此时漏检严重,当 $T_{dmin} = 0.35$ 时,精确度和召回率达到均衡,而 T_{dmax} 的均衡点为0.6。故本文选取参数值为 $T_{IR} = 0.11$, $T_s = 8$, $T_{dmin} = 0.35$, $T_{dmax} = 0.6$ 。

文献[7]算法使用默认参数;文献[8]算法中多尺度参数 σ 取值为(3,4,5,6,7);使用文献[10]算法时,在算法原基础上,将面积大于 3×10^4 的轮廓滤除。

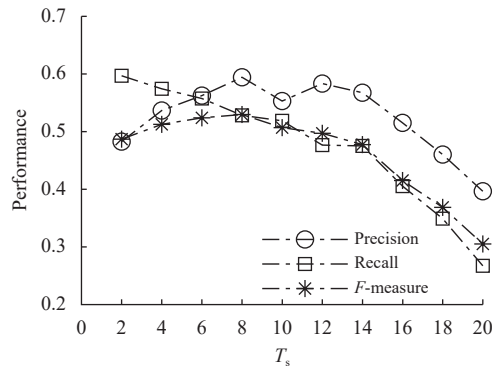
2.2 性能评估

考虑细小裂缝、复杂背景、污渍污染和光照不均匀等情况,本文算法、文献[7]、[8]和[10]算法的部分实验结果分别如图7的3~6列所示,图7中第1列为原图像,第2列为手动标注结果。其他3种算法无法检测特别细的裂缝,在图像背景干扰较多时,无法滤除干扰目标,且分割裂缝的连续性较差,而本文算法能完整检测细小裂缝,且降低复杂背景及不均匀光照干扰,保留裂缝区域。

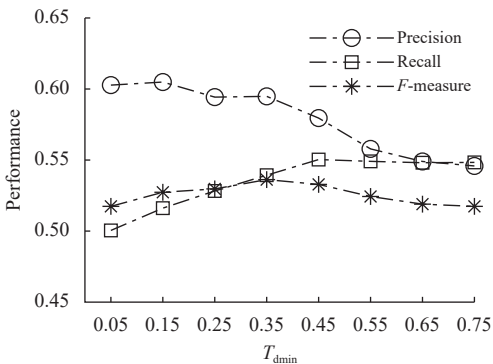
分别统计4种算法的性能指标平均值如表2所示,本文算法的精确率、召回率和综合评价指标F值



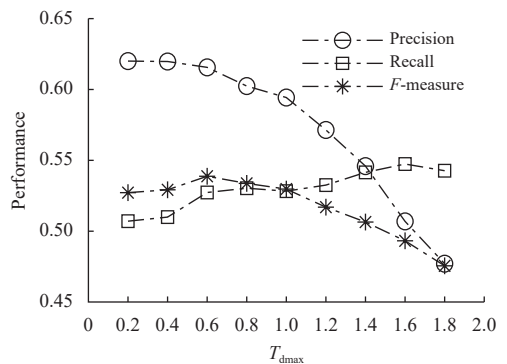
(a) 惯性率阈值灵敏度



(b) 区域生长灰度阈值灵敏度



(c) 短距离阈值灵敏度



(d) 长距离阈值灵敏度

图6 参数灵敏性分析

Fig.6 Parameter sensitivity analysis

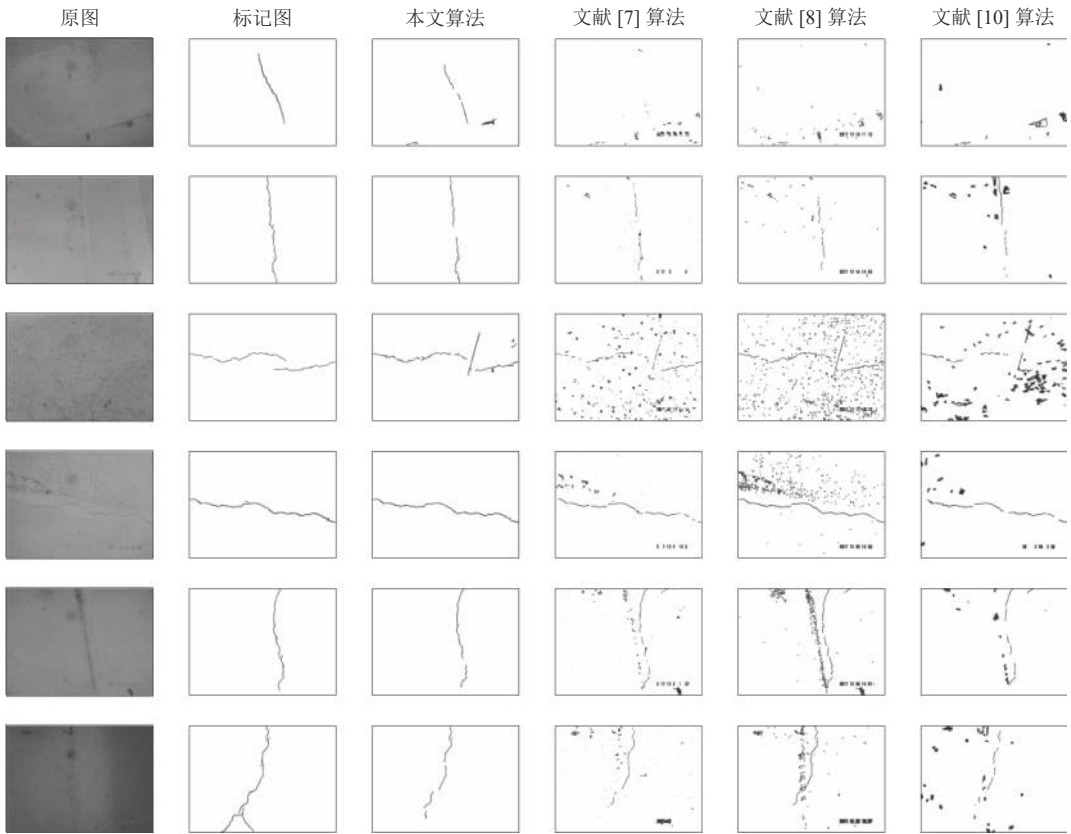


图7 4种算法实验结果

Fig.7 Results of four methods

均大幅度领先于其余3种算法, F 值分别提升115%、148%和238%。在单线程运行速度上, 本文算法处理每张图像的平均耗时仅为1.73 s, 接近文献[10]算法, 远快于文献[7]和文献[8]算法。

表2 4种算法的性能指标值

Table 2 Experimental performance of four methods

算法	精确度	召回率	F 值	运行时间/s
本文算法	0.501	0.613	0.503	1.73
文献[7]	0.191	0.368	0.234	85.2
文献[8]	0.140	0.463	0.203	72.6
文献[10]	0.108	0.308	0.149	1.19

3 结论

本文提出一种嵌入式平台上高分辨率裂缝图像的实时检测算法, 所建立的裂缝聚合模型在一定程度上模拟了裂缝的延伸趋势, 仅将延伸趋势一致的连通域聚合, 抑制了大量灰度表现与裂缝相似的背景干扰。对比实验表明, 本文算法大幅度领先于数种现有算法, 有效提高了桥梁裂缝检测算法的可靠性和实时性, 可应用于嵌入式平台上实时检测桥梁裂缝。但本文算法仅处理灰度图像, 对彩色线状干扰的

抑制能力不足, 后期考虑引入彩色信息, 进一步提高算法的可靠性。

参考文献:

- [1] 张国旗. 基于图像处理的混凝土桥梁底面裂缝检测方法的研究[D]. 北京: 北京交通大学, 2010.
- [2] 王德方, 曾卫明, 王倪传. 基于改进K-means算法的不均匀光照下道路裂缝检测[J]. *计算机应用与软件*, 2015, 32(7): 244-247.
WANG D F, ZENG W M, WANG N C. Road crack detection under uneven illumination using improved k-means algorithm [J]. *Computer Application and Software*, 2015, 32(7): 244-247.
- [3] 张德津, 李清泉, 陈颖, 等. 基于空间聚集特征的沥青路面裂缝检测方法[J]. *自动化学报*, 2016, 42(3): 443-454.
ZHANG D J, LI Q Q, CHEN Y, *et al.* Asphalt pavement crack detection based on spatial clustering feature [J]. *Acta Automatica Sinica*, 2016, 42(3): 443-454.
- [4] 王耀东, 朱力强, 史红梅, 等. 基于局部图像纹理计算的隧道裂缝视觉检测技术[J]. *铁道学报*, 2018, 40(2): 82-90.
WANG Y D, ZHU L Q, SHI H M, *et al.* Vision detection of tunnel cracks based on local image texture calculation [J]. *Journal of the China Railway Society*, 2018, 40(2): 82-90.
- [5] PRASANNA P, DANA K J, GUCUNSKI N, *et al.* Automated crack detection on concrete bridges [J]. *IEEE Trans-*

- actions on Automation Science and Engineering, 2016, 13(2): 591-599.
- [6] ZOU Q, CAO Y, LI Q, *et al.* CrackTree: automatic crack detection from pavement images [J]. *Pattern Recognition Letters*, 2012, 33(3): 227-238.
- [7] 贺福强, 平安, 罗红, 等. 局部特征聚类联合区域增长的桥梁裂缝检测[J]. *科学技术与工程*, 2019, 19(34): 272-277.
HE F Q, PING A, LUO H, *et al.* Bridge crack detection based on local feature clustering combined with regional growth [J]. *Science Technology and Engineering*, 2019, 19(34): 272-277.
- [8] 周飘, 李强, 曾曙光, 等. 基于多尺度Hessian矩阵滤波的陶瓷瓦表面裂纹检测方法[J]. *激光与光电子学进展*, 2020, 57(10): 222-228.
ZHOU P, LI Q, ZENG S G, *et al.* Surface crack detection method for ceramic tile based on hessian matrix multi-scale filtering [J]. *Laser & Optoelectronics Progress*, 2020, 57(10): 222-228.
- [9] CHEN H, ZHAO H, HAN D, *et al.* Accurate and robust crack detection using steerable evidence filtering in electroluminescence images of solar cells [J]. *Optics and Lasers in Engineering*, 2019, 118: 22-33.
- [10] TIAN Q, LUO Q, GE B, *et al.* A methodology framework for retrieval of concrete surface crack's image properties based on hybrid model [J]. *Optik*, 2019, 180: 199-214.
- [11] LIU Z, CAO Y, WANG Y, *et al.* Computer vision-based concrete crack detection using U-net fully convolutional networks [J]. *Automation in Construction*, 2019, 104: 129-139.
- [12] CHEN F C, JAHANSHAHI M R. NB-FCN: real-time accurate crack detection in inspection videos using deep fully convolutional network and parametric data fusion [J]. *IEEE Transactions on Instrumentation and Measurement*, 2019, 69(8): 5325-5334.
- [13] ZHANG L, YANG F, ZHANG Y D, *et al.* Road crack detection using deep convolutional neural network[C]//2016 IEEE international conference on image processing (ICIP). Phoenix: IEEE, 2016: 3708-3712.
- [14] CHAIYASARN K, SHARMA M, ALI L, *et al.* Crack detection in historical structures based on Convolutional Neural Network [J]. *International Journal of GEOMATE*, 2018, 15(51): 240-251.
- [15] KIM B, YUVARAJ N, PREETHAA K R S, *et al.* Surface crack detection using deep learning with shallow CNN architecture for enhanced computation [J]. *Neural Computing and Applications*, 2021, 33(15): 9289-9305.
- [16] SHI W, ZOU R, WANG F, *et al.* A new image segmentation method based on multifractal detrended moving average analysis [J]. *Physica A: Statistical Mechanics and its Applications*, 2015, 432: 197-205.
- [17] 赵玲玲, 汪焯, 刘俊. 基于无人机与HSV空间的光伏电池板检测分析[J]. *红外技术*, 2020, 42(10): 978-982.
ZHAO L L, WANG Y, LIU J. Detection and analysis of photovoltaic panels based on UAV and HSV space [J]. *Infrared Technology*, 2020, 42(10): 978-982.
- [18] 揭云飞, 王峰, 钟有东, 等. 基于地面特征的单目视觉机器人室内定位方法[J]. *广东工业大学学报*, 2020, 37(5): 31-37.
JIE Y F, EVERETT W, ZHONG Y D, *et al.* An indoor positioning method of monocular vision robot based on floor features [J]. *Journal of Guangdong University of Technology*, 2020, 37(5): 31-37.
- [19] 姒绍辉, 胡伏原, 顾亚军, 等. 一种基于不规则区域的高斯滤波去噪算法[J]. *计算机科学*, 2014, 41(11): 313-316.
SI S H, HU F Y, GU Y J, *et al.* Improved denoising algorithm based on non-regular area gaussian filtering [J]. *Computer Science*, 2014, 41(11): 313-316.
- [20] SKLANSKY J. Finding the convex hull of a simple polygon [J]. *Pattern Recognition Letters*, 1982, 1(2): 79-83.
- [21] TOUSSAINT G. Solving geometric problems with the rotating calipers[C]//Proceedings of the IEEE Melecon. Athens: IEEE, 1983.

(责任编辑: 赵少飞)