

基于深度学习与图像处理技术的混凝土表面裂缝智能识别

隆 涛

长安大学理学院, 陕西 西安

收稿日期: 2023年2月17日; 录用日期: 2023年3月13日; 发布日期: 2023年3月21日

摘 要

裂缝检测是保证各类设施安全的关键任务。然而, 由于复杂的结构表面噪声与干扰阻碍裂缝的识别, 深度学习技术识别裂缝后, 无法进一步获得裂缝的相关参数, 裂缝检测仍然具有一定挑战性。为解决上述问题, 提出了基于深度学习与图像处理技术相结合的混凝土表面裂缝智能识别方法。在U-Net基础上, 采用多级特征融合与引入图像梯度构建网络, 实现裂缝的提取。构建包含连通域去噪、断裂连接、边缘检测、裂缝骨架化、裂缝参数计算等流程的图像处理技术, 实现裂缝参数信息自动获取。通过实验验证, 改进的U-net算法提取裂缝, 其mPA达到90.08%。图像处理技术计算裂缝参数, 裂缝宽度与实际值相接近, 相对误差小于10%。研究表明: 本文算法裂缝检测结果精度较高, 且裂缝参数计算精度达到工程应用要求。

关键词

裂缝检测, 深度学习, 图像处理技术, 多级特征融合, 图像梯度

Intelligent Recognition of Concrete Surface Cracks Based on Deep Learning and Image Processing Technology

Tao Long

School of Science, Chang'an University, Xi'an Shaanxi

Received: Feb. 17th, 2023; accepted: Mar. 13th, 2023; published: Mar. 21st, 2023

Abstract

Crack detection is a critical task to ensure the safety of all types of facilities. However, the complex

structural surface noise and interference hinder the identification of cracks. After the deep learning technology identifies the crack, the relevant parameters of the crack cannot be further obtained, and the crack detection is still challenging. In order to solve the above problems, an intelligent identification method for concrete surface cracks based on the combination of deep learning and image processing technology is proposed. On the basis of U-Net, the network is constructed by multi-level feature fusion and the introduction of image gradient to realize the extraction of cracks, and the image processing technology including connected domain denoising, fracture connection, edge detection, fracture skeletonization, fracture parameter calculation and other processes is constructed to realize the automatic acquisition of fracture parameter information. Through experimental verification, the improved U-net algorithm can extract cracks and make the mPA reach 90.08%. The image processing technology calculates the crack parameters, and the crack width is close to the actual value, and the relative error is less than 10%. The research results show that the algorithm in this paper has high accuracy of crack detection results, and the calculation accuracy of crack parameters meets the requirements of engineering application.

Keywords

Crack Detection, Deep Learning, Image Processing Technology, Multi-Level Feature Fusion, Image Gradient

Copyright © 2023 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

裂缝是混凝土结构建筑诸多损伤中最为常见的一种损伤表现，它是结构现场安全性检测监测以及模型试验研究的重要指标。在工程结构现场检测与实验室实验中，目前人工检测仍然是裂缝检测的主要方法。然而，人工检测需要维护人员进行现场勘查、标记、测量，并记录结果。实践表明，人工检测裂缝主观性强、工作强度大、效率低且存在安全隐患，因此人工检测裂缝无法满足工程实践的需求。

近年来，随着数字图像处理技术的发展，利用数字图像处理技术进行结构表面裂缝检测成为有效的检测方法。根据 Zaker 等人[1]的归纳，与裂缝检测相关研究与应用相对较多的有 4 种，分别基于阈值、边缘、区域、匹配。其中 Sobel 算子以及 Canny 算子在裂缝检测中应用最为广泛[2]。与传统人工检测相比，虽然裂缝的提取过程更加客观，实现了裂缝形态和参数的准确记录，但图像处理过程复杂，且易受环境影响。

随着深度学习的兴起，裂缝检测领域研究逐步由传统的图像处理技术转变为基于深度学习方法进行检测。通过对大量样本进行学习，自动提取裂缝特征，实现裂缝的识别与提取。Hsieh 等人[3]根据不同目的将基于深度学习的裂缝检测算法分成了 3 类：图像分类算法、目标检测算法、图像分割算法。Cha 等人[4]首次利用 CNN 模型来检测混凝土表面裂缝，结果表明 CNN 优于传统裂缝检测方法。与分类算法不同，目标检测算法不仅需要判断类别，而且需要定位裂缝的位置。目标检测算法可分为以 RCNN [5]为代表的二阶段算法和以 YOLO [6]为代表的一阶段算法。无论分类算法还是目标检测算法，都无法进行后续裂缝特征参数计算，而图像分割算法则能实现后续裂缝特征参数计算。常用的图像分割算法包括 FCN [7]、SegNet [8]、PSPNet [9]、DeepLab [10]、U-Net [11]等，Ren 等人[12]提出了基于 SegNet 的裂缝检测网络 CrackSegNet；于海洋等人[13]改进 U-Net 模型，建立了一种融合残差和注意力机制的道路裂缝检测

模型。以 U-Net 为代表的编码 - 解码结构深度学习网络模型，其结构适用于细长物体检测，常用于裂缝检测领域。由于实际结构表面裂缝复杂多样，目前裂缝检测也存在着一些难题：预测结果边缘较厚，裂缝轮廓定位不精准；深度学习技术在提取裂缝后，无法更进一步获得裂缝的相关参数。

为了解决上述问题，本文采用深度学习算法与图像处理技术相结合来识别裂缝的形态和计算裂缝几何特征，改进 U-Net 深度学习算法来提取裂缝形态，后续利用连通域去噪、断裂连接、边缘检测、裂缝细化等图像处理技术来计算裂缝的几何特征，通过实验验证本文检测方法的可行性。

2. 混凝土表面裂缝检测方法

混凝土表面裂缝检测方法主要步骤包括移动智能手机裂缝图像采集、像素标定、基于深度学习算法的裂缝提取、裂缝参数测量。

2.1. 裂缝图像采集

利用移动智能手机对校园里一段混凝土路面进行裂缝图像采集，本次采用如图 1 所示的设备进行拍摄，用三角支架固定手机，使手机平面与地面平行。为方便后续计算，固定拍摄距离大小，通过多次尝试，最终固定手机与地面距离为 40 cm 为最佳拍摄距离。



Figure 1. Schematic diagram of crack image acquisition device

图 1. 裂缝图像采集装置示意图

2.2. 像素标定

物体的像素尺寸并非实际尺寸，要将图像中的像素尺寸转换为实际尺寸，需要对像素进行标定。本文采用二维码技术[14]进行像素标定。如图 2 所示，将待测裂缝旁放置标准尺寸二维码贴纸作为标识，二维码采用统一尺寸，AB、AC 的实际尺寸为 25 mm。利用 python 软件获得 AB 的像素长度 d_1 ，AC 的像素长度 d_2 ，理想情况下， $AB = AC$ 。为减小手机平面与地面倾斜带来的拍摄误差，取两直角边均值为

$$d = \frac{d_1 + d_2}{2} \quad (1)$$

因此可根据实际二维码尺寸与其所成像的像素尺寸获得像素标定值，公式如下：

$$k = \frac{D}{d} \quad (2)$$

其中， D 为二维码的实际尺寸，本文为 25 mm； d 为二维码图像的像素尺寸，单位为 pixel。



Figure 2. Schematic diagram of QR code placement

图 2. 二维码放置示意图

2.3. 深度学习算法与实现

2.3.1. 裂缝分割整体框架

U-Net 是编码-解码结构的图像分割算法。编码阶段即特征提取阶段，结构与 VGG16 相似，由卷积与池化堆叠而成。解码阶段即加强特征提取阶段，将编码阶段最底层特征逐步上采样还原，与编码阶段特征层进行融合，最终还原到原始图像分辨率大小。U-Net 上采样恢复图像分辨率过程中，不仅丢失很多裂缝特征信息，而且还会导致裂缝边缘定位不精准。于是本文在 U-Net 不足上进行改进。如图 3 所示，在 U-Net 基础上进行多级特征融合，防止特征信息丢失。另外增加一个边缘细化分支，用于定位裂缝边缘。

采用多级特征融合，防止特征丢失。将编码阶段五个加强特征层 P1~P5 进行特征拼接。融合的特征层包含了丰富的细节特征信息的同时，还包含丰富的高级语义信息。

边缘细化分支对预测结果进行边缘细化定位。Canny 算子的边缘检测能够得到真实的裂缝边缘特征。本文采用 Canny 算子得到梯度图像，将梯度图像与多级融合特征层相结合，以细化边缘预测结果。

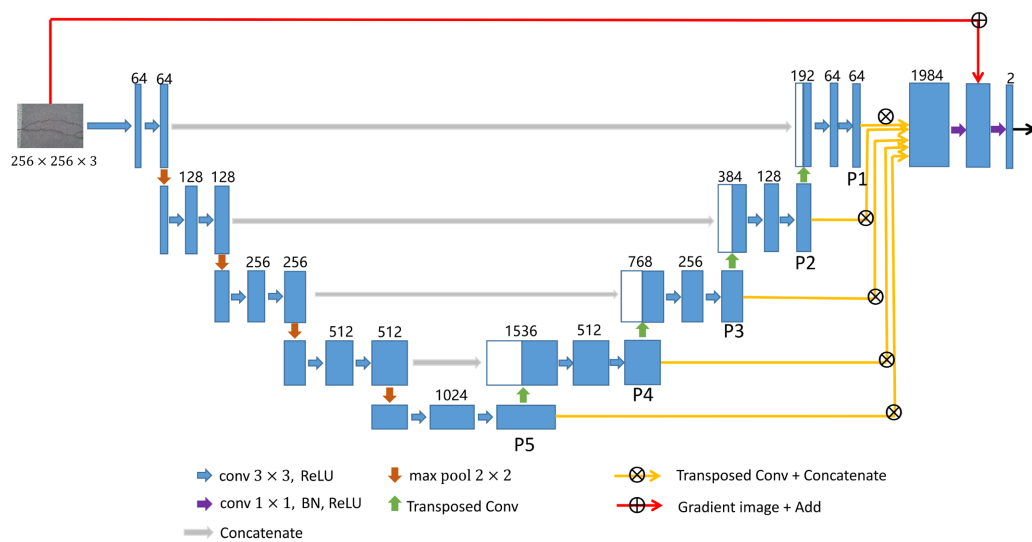


Figure 3. Structure of crack segmentation network

图 3. 裂缝分割网络结构图

2.3.2. 损失函数

裂缝检测是一个二分类任务，常用交叉熵作为损失函数。对于裂缝图像来说，正负样本极不平衡，使用交叉熵损失函数进行网络训练，网络更偏向于背景预测。为降低样本不平衡带来的影响，本文采用加权交叉熵损失函数[15]，定义如下：

$$L_{WCE} = -\frac{1}{N} \sum_{i=1}^N [\beta y_i \log \hat{y}_i + (1-\beta)(1-y_i) \log(1-\hat{y}_i)] \quad (3)$$

其中， N 表示像素数， y_i 表示第*i*个像素点的标签， \hat{y}_i 表示第*i*个像素点的预测结果， $\beta = |G^-| / (|G^+| + |G^-|)$ ， $|G^+|$ 和 $|G^-|$ 分别表示裂缝和非裂缝像素数量。

加权交叉熵函数关注的是每个像素的预测情况，没有考虑裂缝预测的整体情况。因此在加权交叉熵函数的基础上引入 Dice 损失函数[16]，定义如下：

$$L_{Dice} = 1 - \frac{2 \sum_{i=1}^N y_i \cdot \hat{y}_i + \varepsilon}{\sum_{i=1}^N y_i^2 + \sum_{i=1}^N \hat{y}_i^2 + \varepsilon} \quad (4)$$

其中，参数 ε 设置为 1，防止分母为 0 情况。

因此，本文网络模型训练的总损失函数为：

$$L = L_{WCE} + L_{Dice} \quad (5)$$

2.4. 裂缝参数测量

通过深度学习算法提取裂缝后，采用数字图像处理技术对预测结果进行后处理，计算裂缝的宽度，具体步骤包括连通域去噪、断裂连接、边缘检测、裂缝骨架化、裂缝宽度计算等，如图 4 所示。

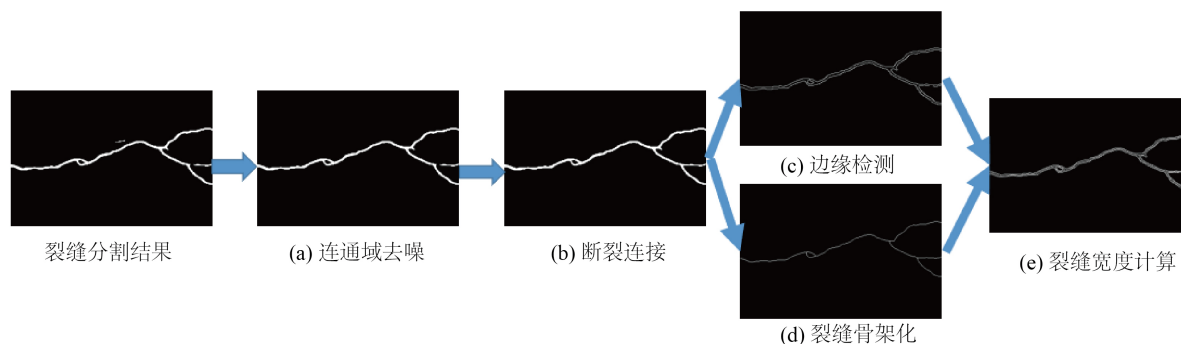


Figure 4. Measurement process of crack parameters

图 4. 裂缝参数测量过程

2.4.1. 连通域去噪

如图 4(a)所示，首先采用八连通域进行标记，然后计算各标识块的像素数，最后删除像素数小于阈值的连通域。

2.4.2. 断裂连接

基于图像分割的算法预测裂缝，可能出现断裂的情况，断裂处会直接影响后续裂缝宽度的计算。传统的形态学方法处理断裂，可能破坏裂缝的连通性和完整的骨架结构。为了获得完整、连续的裂缝，采用基于 KD-tree 的裂缝连接算法[17]对分割后的裂缝进行连接，结果如图 4(b)所示。

2.4.3. 边缘检测

如图 4(c)所示, 采用 Canny 算子进行边缘检测。对图像进行灰度化, 利用高斯滤波来平滑图像去除噪声, 计算梯度幅值和方向, 对梯度幅值进行非极大值抑制来消除边误检, 用双阈值算法来寻找可能的边界, 利用滞后技术来跟踪边界。

2.4.4. 裂缝骨架化

本文采用 Zhang-Suen 并行细化算法[18]来提取裂缝的骨架, 如图 5 所示, 中心像素点 P1 的八邻域图, 具体步骤如下:

步骤 1: 循环所有的裂缝像素并将同时满足以下条件的裂缝像素转为背景像素:

- a) $2 < N(P1) < 6$
- b) $S(P1) = 1$
- c) $P2 \times P4 \times P6 = 0$
- d) $P4 \times P6 \times P8 = 0$

步骤 2: 满足以下全部条件的像素点也从裂缝像素转为背景像素:

- a) $2 < N(P1) < 6$
- b) $S(P1) = 1$
- c) $P2 \times P4 \times P8 = 0$
- d) $P2 \times P6 \times P8 = 0$

其中, $N(P1)$ 表示 P1 中心点周围属于裂缝的像素点的总数, $S(P1)$ 表示 P2~P9~P2 顺时针 0~1 累积出现的次数。

P9	P2	P3
P8	P1	P4
P7	P6	P5

Figure 5. Eight fields of P1 pixel

图 5. P1 像素点的八邻域

使用 Zhang-Suen 细化算法, 可以得到裂缝的骨架, 但提取的骨架还存在一些毛刺、假枝和非单像素宽度问题, 如图 6 所示。这些问题会影响后续裂缝参数的计算, 因此需要对细化后的结果再进行处理。

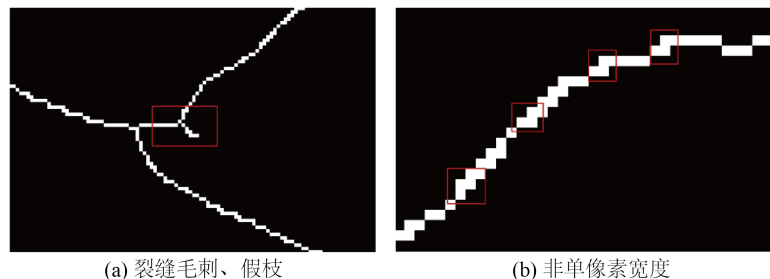


Figure 6. Display of thinning algorithm problems

图 6. 细化算法问题展示

非单像素宽度问题指的是非分支点处的连通性大于 2。经统计在裂缝骨架上会出现如图 7 所示的四种非单像素宽度情况。本文采用基于模板匹配的方法来解决非单像素宽度问题。具体步骤为：遍历裂缝骨架，若骨架上的裂缝像素点 P1 及其八邻域像素满足下面任意一种情况，则将该裂缝像素点 P1 转化为背景像素点。具体情况为：

- 1) $P1 = P2 = P8 = 1$ 且 $P4 = P5 = P6 = 0$
- 2) $P1 = P2 = P4 = 1$ 且 $P6 = P7 = P8 = 0$
- 3) $P1 = P4 = P6 = 1$ 且 $P2 = P8 = P9 = 0$
- 4) $P1 = P6 = P8 = 1$ 且 $P2 = P3 = P4 = 0$

按照上述算法，可实现裂缝骨架的单像素宽化。

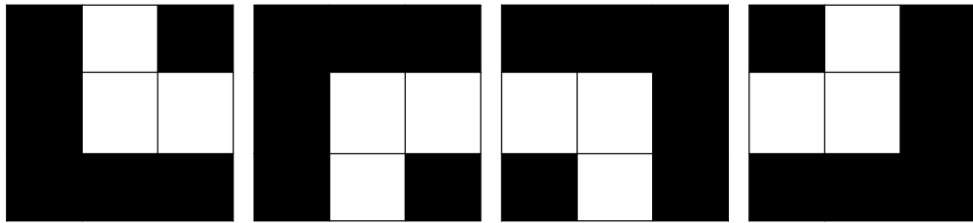


Figure 7. Non-single-pixel width
图 7. 非单像素宽情况

针对裂缝骨架中的毛刺、假枝，本文基于连通域标记方法来进行消除，具体步骤如下：

- 1) 遍历单像素宽化后的裂缝骨架每一像素点，直到所有裂缝像素点被遍历完毕；
- 2) 统计当前点 P1 的八邻域中为裂缝像素的个数，若个数大于 2 则表明点 P1 是分支点，记录其 P1 位置并将 P1 像素值设置为 0；
- 3) 将 P1 转化为背景像素后，此时图像中会出现多个连通域，分别计算每个连通域的像素和；
- 4) 设置一个阈值 T，删除连通域像素个数和小于 T 的连通域，即从裂缝像素转化成背景像素，反正，不作处理；
- 5) 统计点 P1 的八邻域范围内的裂缝像素个数，若个数大于 1 则将点 P1 像素值从 0 变为 1，反正，不作处理。继续遍历下个裂缝像素。

通过设定阈值，可以消除掉裂缝骨架中较小的毛刺、假枝，同时也能够消除骨架图像中的孤立点。

2.4.5. 裂缝宽度计算

裂缝宽度的大小代表着路面的破损等级，是评价裂缝危害等级的一个重要标准。本文使用正交骨架线法[19]来计算裂缝的宽度，如图 8 所示，两条黑线代表裂缝边缘轮廓，红线为裂缝骨架线，裂缝宽度计算步骤如下：

步骤 1：计算骨架线上任意像素点 P_n 的法线梯度。

为了计算 P_n 点处的法线梯度，首先需要计算 P_n 与之相邻的两个裂缝像素点 P_{n-1} 、 P_{n+1} 拟合的抛物线函数： $y = ax^2 + bx + c$ ，然后求 P_n 点处在抛物线上的一阶偏导数，即 P_n 的斜率 $m_{T(P_n)}$ 。 P_n 的斜率与法线梯度互为负倒数，于是 P_n 的法线梯度可由公式(6)计算出：

$$m_{N(P_n)} = -\frac{1}{m_{T(P_n)}} \tag{6}$$

步骤 2：根据 P_n 处的法线长度计算该点的裂缝宽度。

P_n 的法线 $m_{N(P_n)}$ 与裂缝边缘相交的两点距离即为 P_n 处的裂缝宽度。如图 8 中，计算两点之间的距离即红色直线便是点 P_n 的裂缝宽度。

步骤 3：计算骨架上每个像素点的裂缝宽度，得到裂缝宽度的平均值和最大值。

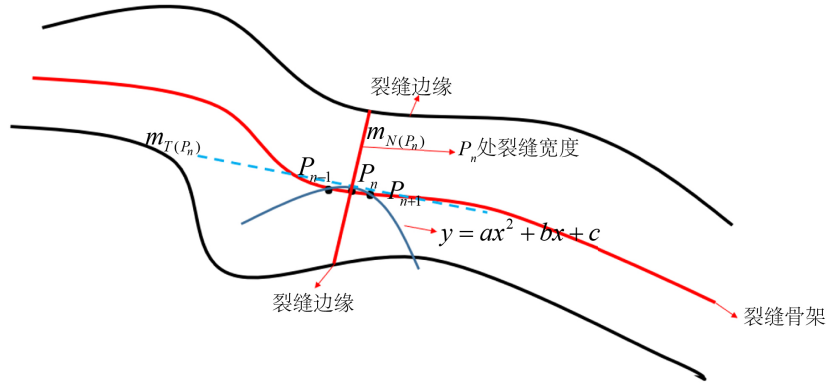


Figure 8. Schematic diagram of crack width calculation
图 8. 裂缝宽度计算示意图

3. 实验结果与分析

3.1. 评价指标

为了评估我们的工作，我们在分割任务中引入了常见的评估指标[20]：Precision, Recall, F1-score, PAM, PA 和 MIou, 其公式如下：

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad (7)$$

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

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

$$\text{PA} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \quad (10)$$

$$\text{MPA} = \frac{1}{2} \left(\frac{\text{TP}}{\text{TP} + \text{FN}} + \frac{\text{TN}}{\text{FP} + \text{TN}} \right) \quad (11)$$

$$\text{MIoU} = \frac{1}{2} \left(\frac{\text{TP}}{\text{TP} + \text{FN} + \text{FP}} + \frac{\text{TN}}{\text{TN} + \text{FN} + \text{FP}} \right) \quad (12)$$

其中，TP、FP、TN、FN 分别表示真阳性、假阳性、真阴性、假阴性。

3.2. 裂缝分割实验结果与分析

本文基于 python 的深度学习框架 tensorflow2 上进行实验。模型训练和预测在 Linux 系统上完成。训练过程中，设置早期停止，批处理大小设置为 16，训练周期设置为 200，采用 Adam 优化器优化网络模型。

表 1 给出了算法的评价指标结果，与 U-Net 相比，改进后的算法检测精度更高，其平均精度达到了 90.08%，验证了改进后的算法的有效性和优越性。

Table 1. Quantitative evaluation results of algorithm
表 1. 算法定量评价结果

算法	PA	MPA	MIoU	Precision	Recall	F1-score
U-Net	72.73	86.18	78.90	86.90	86.35	86.58
Ours	80.71	90.08	79.26	84.56	90.06	87.07

本文算法进行训练后，在测试集中进行裂缝检测，部分结果如图 9 所示，可以看出本文提出的裂缝分割算法能够完整地识别出裂缝。

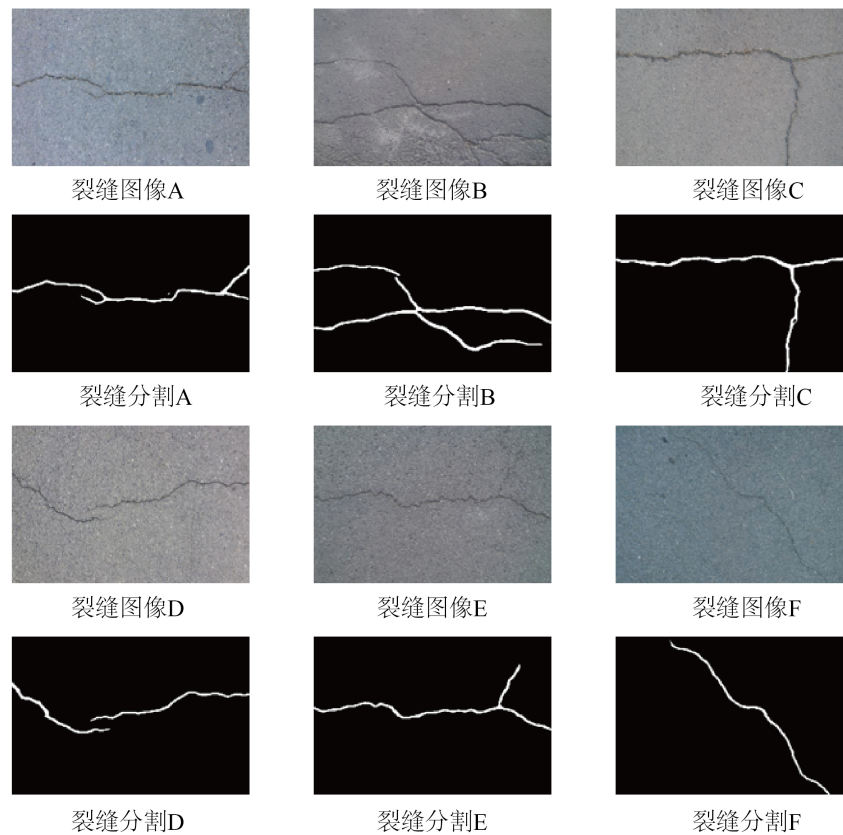


Figure 9. Crack identification results of the algorithm in this paper
图 9. 本文算法裂缝识别结果

3.3. 裂缝宽度检测

为验证本文裂缝宽度计算方法的有效性，对图 9 中的裂缝识别结果进行宽度检测，并使用裂缝测宽仪测量裂缝真实值，测量精度为 0.01 mm。为直观显示裂缝宽度的变化情况，对图 9 裂缝分割 F 沿裂缝骨架的方向计算每一个裂缝测量点的裂缝宽度，得到了裂缝宽度分布曲线，如图 10 所示。

采用本文宽度测量方法，计算图 9 裂缝图像裂缝宽度最大值和平均值，裂缝宽度计算结果如表 2 所示。从表中可以看出，本文计算裂缝宽度结果与实测值总体差距不大，检测结果与实测结果最大宽度和平均宽度相对误差都 10% 以内，证明本文宽度检测方法的有效性。

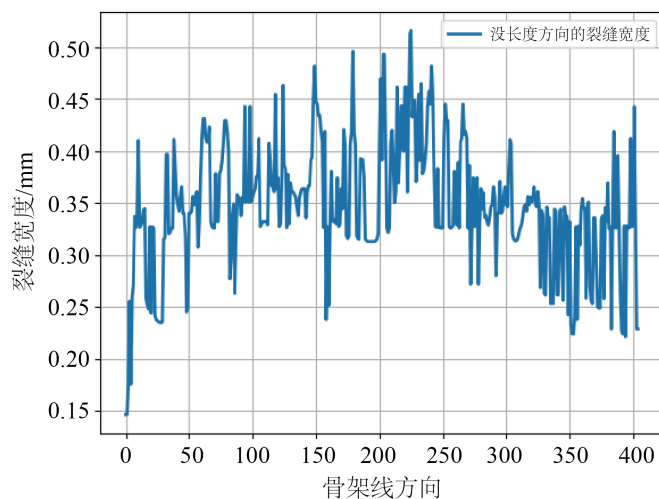


Figure 10. Crack width distribution curve

图 10. 裂缝宽度分布曲线

Table 2. Crack width test results

表 2. 裂缝宽度检测结果

图片编号	实测宽度/mm		计算宽度/mm		绝对误差/mm		相对误差/(%)	
	最大值	平均值	最大值	平均值	最大值	平均值	最大值	平均值
A	1.15	0.38	1.177	0.408	0.027	0.028	2.35	7.37
B	1.27	0.36	1.316	0.389	0.046	0.029	3.62	8.06
C	0.96	0.46	0.998	0.482	0.038	0.022	3.96	4.78
D	0.78	0.40	0.767	0.379	0.013	0.021	1.67	5.25
E	0.92	0.39	0.941	0.401	0.021	0.011	2.28	2.82
F	0.52	0.35	0.548	0.370	0.028	0.020	5.38	5.71

4. 结论

针对现有裂缝检测研究上面的不足，本文提出了一种基于深度学习与图像处理技术相结合的混凝土表面裂缝智能识别方法，并进行了实验验证。

1) 对 U-Net 深度学习图像分割算法不足进行改进来提取裂缝。采用多级特征融合，防止特征信息丢失，添加边缘细化分支对预测结果进行边缘细化定位。

2) 为提高裂缝参数检测精度，深度学习算法提取裂缝后进行了一系列的图像处理技术，宽度计算结果与真实值对比显示：裂缝宽度最大值与平均值相对误差小于 10%，属于误差范围内，验证了本方法的有效性与可行性。

3) 在后续工作中可以考虑利用深度学习的方法来计算裂缝的宽度，简化计算量的同时也避免背景复杂度的影响。

参考文献

- [1] Zakeri, H., Nejad, F.M. and Fahimifar, A. (2017) Image Based Techniques for Crack Detection, Classification and Quantification in Asphalt Pavement: A Review. *Archives of Computational Methods in Engineering*, **24**, 935-977. <https://doi.org/10.1007/s11831-016-9194-z>

- [2] Sattar, D., Thomas, R.J. and Marc, M. (2018) Comparison of Deep Convolutional Neural Networks and Edge Detectors for Image-Based Crack Detection in Concrete. *Construction and Building Materials*, **186**, 1031-1045. <https://doi.org/10.1016/j.conbuildmat.2018.08.011>
- [3] Hsieh, Y.A. and Tsai, Y.J. (2020) Machine Learning for Crack Detection: Review and Model Performance Comparison. *Journal of Computing in Civil Engineering*, **34**, Article ID: 04020038. [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000918](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000918)
- [4] Cha, Y.J., Choi, W. and Buyukozturk, O. (2017) Deep Learning-Based Crack Damage Detection Using Convolutional Neural Networks. *Computer-Aided Civil and Infrastructure Engineering*, **32**, 361-378. <https://doi.org/10.1111/mice.12263>
- [5] Ren, S., He, K., Girshick, R., et al. (2017) Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, **39**, 1137-1149. <https://doi.org/10.1109/TPAMI.2016.2577031>
- [6] Zhang, Y., Huang, J. and Cai, F. (2020) On Bridge Surface Crack Detection Based on an Improved YOLO v3 Algorithm. *IFAC-PapersOnLine*, **53**, 8205-8210. <https://doi.org/10.1016/j.ifacol.2020.12.1994>
- [7] Lecun, Y., Bengio, Y. and Hinton, G. (2015) Deep Learning. *Nature*, **521**, 436-444. <https://doi.org/10.1038/nature14539>
- [8] Badrinarayanan, V., Kendall, A. and Cipolla, R. (2017) SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **39**, 2481-2495. <https://doi.org/10.1109/TPAMI.2016.2644615>
- [9] Zhao, H., Shi, J., Qi, X., et al. (2017) Pyramid Scene Parsing Network. 2017 *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, 25-27 April 2017, 6230-6239. <https://doi.org/10.1109/CVPR.2017.660>
- [10] Chen, L.C., Papandreou, G., Kokkinos, I., et al. (2017) Deeplab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected Crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **40**, 834-848. <https://doi.org/10.1109/TPAMI.2017.2699184>
- [11] Ronneberger, O., Fischer, P. and Brox, T. (2015) U-Net: Convolutional Networks for Biomedical Image Segmentation. *18th International Conference*, Munich, 5-9 October 2015, 234-241. https://doi.org/10.1007/978-3-319-24574-4_28
- [12] Ren, Y., Huang, J., Hong, Z., et al. (2020) Image-Based Concrete Crack Detection in Tunnels Using Deep Fully Convolutional Networks. *Construction and Building Materials*, **234**, 117367. <https://doi.org/10.1016/j.conbuildmat.2019.117367>
- [13] 于海洋, 景鹏, 张文涛, 等. 基于残差和注意力机制的道路裂缝检测 U-Net 改进模型[J]. *计算机工程*, 2022, 1-14.
- [14] 李红卫, 熊韬. 基于二维码识别的 P4P 算法研究[J]. *激光与光电子学进展*, 2020, 57(12): 304-312.
- [15] Ren, X., Xing, Z., Xia, X., Grundy, et al. (2019) Neural Network-Based Detection of Self-Admitted Technical Debt: From Performance to Explainability. *ACM Transactions on Software Engineering and Methodology*, **28**, 15.1-15.45. <https://doi.org/10.1145/3324916>
- [16] Eelbode, T., Bertels, J., Berman, M., et al. (2020) Optimization for Medical Image Segmentation: Theory and Practice When Evaluating with Dice Score or Jaccard Index. *IEEE Transactions on Medical Imaging*, **39**, 3679-3690. <https://doi.org/10.1109/TMI.2020.3002417>
- [17] 陈瑶. 基于图像分析的桥梁裂缝检测方法研究[D]: [硕士学位论文]. 合肥: 中国科学技术大学, 2016.
- [18] Zhang, T.Y. and Suen, C.Y. (1984) A Fast Parallel Algorithm for Thinning Digital Patterns. *Communications of the ACM*, **27**, 236-239. <https://doi.org/10.1145/357994.358023>
- [19] Grelard, F., Baldacci, F., Vialard, A., et al. (2017) New Methods for the Geometrical Analysis of Tubular Organs. *Medical Image Analysis*, **42**, 89-101. <https://doi.org/10.1016/j.media.2017.07.008>
- [20] Fu, H.X., Meng, D., Li, W.H., et al. (2021) Bridge Crack Semantic Segmentation Based on Improved Deeplabv3+. *Journal of Marine Science and Engineering*, **9**, 671. <https://doi.org/10.3390/jmse9060671>