复杂背景下基于图像处理的桥梁裂缝检测算法

李良福**, 孙瑞赟*
陕西师范大学计算机科学学院, 陕西 西安 710119

摘要 针对传统桥梁裂缝检测算法不能准确提取裂缝的问题,提出了一种复杂背景下基于图像处理的桥梁裂缝检测算法。根据深度卷积生成式对抗网络原理,利用桥梁裂缝图像生成模型,对数据集进行扩增。针对裂缝特征构建基于语义分割的桥梁裂缝图像分割模型,利用桥梁裂缝图像分割模型提取高分辨率裂缝图像中的裂缝。研究结果表明,与现有算法相比,所提算法在复杂道路场景中具有更好的检测效果和更强的泛化能力。

关键词 图像处理; 复杂背景; 桥梁裂缝检测; 深度卷积生成式对抗网络; 语义分割
中图分类号 TP391.9 **文献标识码** A **doi**: 10.3788/LOP56.061002

Bridge Crack Detection Algorithm Based on Image Processing under Complex Background

Li Liangfu**, Sun Ruiyun*
School of Computer Science, Shaanxi Normal University, Xi'an, Shaanxi 710119, *China*

Abstract In order to solve the problem that the traditional bridge crack detection algorithm cannot extract cracks accurately, a bridge crack detection algorithm is proposed based on image processing, which is suitable for complex scenes. According to the principle of the deep convolutional generative adversarial network, the bridge crack image generative model is proposed and used to amplify the dataset. For the characteristics of bridge cracks, a bridge crack image segmentation model is constructed based on semantic segmentation. The bridge crack image segmentation model is used to extract the bridge cracks from the high-resolution crack images. The research results show that the proposed algorithm has a better detection effect and a stronger generalization ability in the complex road scenes compared with the existing algorithms.

Key words image processing; complex background; bridge crack detection; deep convolutional generative adversarial network; semantic segmentation
OCIS codes 100.2000; 100.4996; 100.1830; 150.1135

1 引 言

交通运输是经济发展的基本需要和先决条件,是现代社会的生存基础和文明标志,是工业发展的基础设施和重要纽带,关系着国家经济的发展,承载着社会进步的命脉[1-2]。据《2017 年国民经济和社会发展统计公报》[3]统计,在 2017 年,我国新改建高速公路里程为 6796 km,新建高速铁路投产里程为 2182 km,建成了世界上最大的高速公路网和高铁运营网。我国重建设而忽养护的公路桥梁管理模式,加快了桥梁与道路老化的速度,容易造成安全隐患[4]。已有研究表明,混凝土桥梁长期受直接应力与次生应力的作用将会产生桥梁裂缝,而桥梁裂缝是桥梁损伤状况的外在表现,是桥梁结构达到承载力极限的标志。因此,对混凝土桥梁裂缝的检测十分重要。

采取有效手段对桥梁裂缝进行检测[5]对确保公共交通的安全和正常运行具有十分重要的作用,长

收稿日期: 2018-09-12; **修回日期**: 2018-09-25; **录用日期**: 2018-09-30
基金项目: 国家自然科学基金(61573232, 61401263)
* **E-mail**: 984789463@qq.com; ** **E-mail**: longford@xjtu.edu.cn

期以来受到了国内外学术界、工程界的广泛关注，并且取得了一些优秀的研究成果。Oh 等[6]提出了迭代的阈值分割算法，但该算法的阈值需要人工设置；孙亮等[7]提出了基于自适应阈值 Canny 算法的裂缝检测方法，该算法解决了需要人工设置阈值的缺陷，但没有考虑光照不均与噪声对裂缝识别造成的影响，很难保证稳定性；Talab 等[8]利用 Otsu 与多重滤波结合的算法提取裂缝，该算法考虑了由光照不均造成的影响，但是单一的阈值使得该算法并不能适用于所有的图像。

近年来，机器学习与深度学习成为人工智能领域飞速发展的热点，许多科研工作者成功地将桥梁裂缝检测与之结合。Zhang 等[9]利用深度卷积神经网络实现对桥梁裂缝的提取与检测，但是该算法检测到的裂缝宽度与真实裂缝宽度相差极大；Chen 等[10]使用卷积神经网络和朴素贝叶斯数据融合的 NB-CNN 网络进行裂缝检测，该算法的优点是可以检测到微小的裂缝，但仅能检测到裂缝的位置，并不能实现对裂缝的提取；Shi 等[11]利用随机结构森林算法对裂缝进行自动提取，但是当该算法应用于复杂背景时，检测效果欠佳。以上算法取得的良好实验效果均建立在裂缝图像的背景简单且不存在任何障碍物的情形之下，低估了桥梁路面的复杂程度。基于此，本文提出了一种基于图像处理且适用于复杂背景的桥梁裂缝检测算法。具体做法为：首先，通过桥梁裂缝图像生成模型扩增数据集，接着，利用扩增的数据集训练桥梁裂缝图像分割模型，最后，利用桥梁裂缝图像分割模型检测高分辨率裂缝图像中的裂缝。

2 数据集的扩增

这里提出的复杂背景下桥梁裂缝检测算法属于深度学习领域，众所周知，深度学习中的网络模型所涉及的权重参数众多，在训练样本不足的情况下很容易出现欠拟合。因此，必须建立有充足样本的数据集。但是到目前为止，全球没有公开的、适用于复杂背景研究的桥梁裂缝数据集。而如果直接用人工方式大量采集，不仅工作量极大，而且效率极低。目前，由 Radford 等[12]提出的深度卷积生成式对抗网络(DCGAN)是人工智能领域的一个研究热点，已经被成功用于生成全新图像。但文献[12]中的 DCGAN 模型适用于生成 64 pixel×64 pixel 大小、类似 cifar-10、SVHN 数据集的图像，无法有效生成 256 pixel×256 pixel、具有线性拓扑结构的桥梁裂缝的图像。因此，提出了一种基于深度卷积生成式对抗网络的桥梁裂缝图像生成模型，简称 BCIGM。

2.1 采集数据扩增

使用 BCIGM 需要几千张图像作为训练样本，如果全部人工采集仍然是不易实现的。因此，数据集的扩增分两步完成：第一步，对少量的数据通过图像的几何变换、空间滤波、线性变换这 3 类图像处理方法进行扩充；第二步，使用 BCIGM 生成大量裂缝图像。经过第一步数据集扩充之后，人工手动挑选出 9362 幅图像作为 BCIGM 的训练集，经过图像处理算法扩充之后，部分桥梁裂缝图像数据集扩增示意图如图 1 所示。

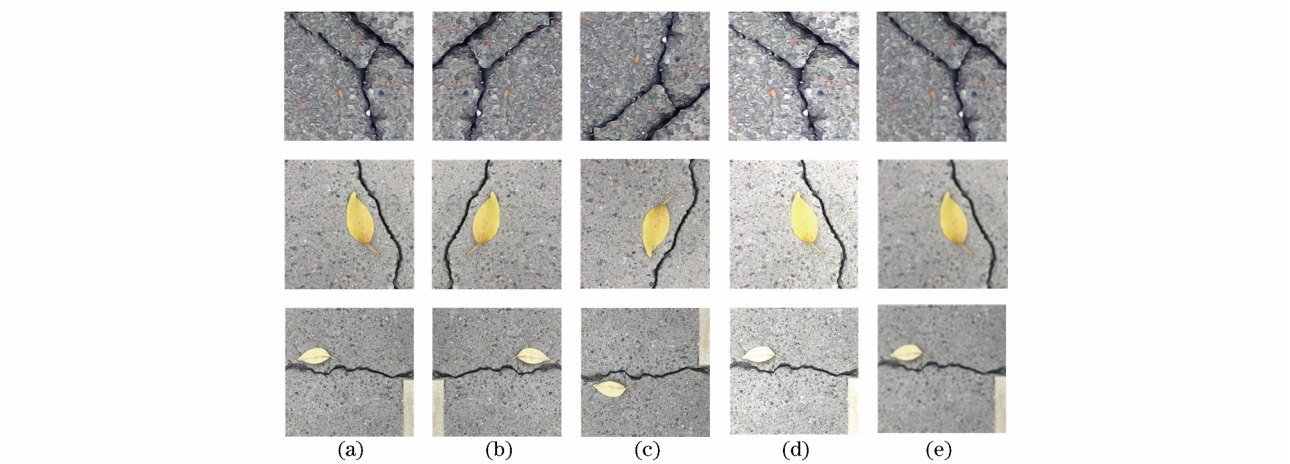

图 1 桥梁裂缝图像数据集扩增示意图。(a)原图；(b)水平翻转；(c)垂直翻转；(d)线性变换；(e)空间滤波变换

Fig. 1 Schematic of dataset amplification of bridge crack images. (a) Original image; (b) horizontal flip; (c) vertical flip; (d) linear transformation; (e) spatial filtering transformation

2.2 桥梁裂缝图像生成模型

2.2.1 模型原理

桥梁裂缝图像生成模型的原理与生成式对抗网络的原理一样，由具有对抗关系的生成模型(G)与判别模型(D)组成。其中，生成模型(G)的作用是通过学习已有真实数据样本的概率分布，生成尽可能服从真实样本分布的样本 $G(z)$，判别模型(D)的作用是判断输入来自真实样本还是来自生成样本 $G(z)$，实质为一个二分类模型。训练 BCIGM 的过程就是不断交替更新 G 和 D 的过程，当生成网络 G 固定时，优化判别网络，使输入真实数据样本时的输出尽可能趋向 1，输入生成数据样本时的输出尽可能趋向 0；当判别网络(D)固定时，优化生成网络(G)，使生成样本经过判别网络(D)后输出高概率，当双方达到纳什均衡时，生成模型(G)生成了判别模型(D)无法判断是否是真实样本的生成样本。上述过程可表示为

$$\min_{G}\max_{D}V(\mathrm{D},\mathrm{G})=E_{x\sim p_{\mathrm{data}(x)}}[\log_a D(x)]+E_{z\sim p_{z(z)}}\{\log_a\{1-D[G(z)]\}\},\tag{1}$$

式中：z 为输入生成网络的噪声；$G(z)$ 为生成样本；x 为真实数据样本；a 为大于 0 且不等于 1 的任意底数；$D(x)$ 为 x 通过判别网络后判断为真实样本的概率；$D[G(z)]$ 为生成样本通过判别网络后判断为真实样本的概率；$P_{\mathrm{data}(x)}$ 和 $P_{z(z)}$ 分别为真实样本数据的概率分布和初始噪音数据的概率分布；$E(\cdot)$ 表示计算期望值。

2.2.2 生成模型

生成模型的作用是将输入的噪声通过上采样生成 256 pixel×256 pixel×3 的裂缝图像，主要由转置卷积构成。具体过程为：首先，输入 100 维的噪声；接着，通过全连接层后进行维度转换，转换为 1024 个 8×8 的特征图；最后，通过 5 个卷积核为 5×5，步幅为 2，卷积核数目依次为 512，256，128，64，3 的转置卷积，且除最后一个转置卷积外，其余转置卷积后均使用 SeLU 激活函数[13]。生成模型如图 2 所示。

一般来说，转置卷积是由深而窄的层次延展为窄而深的层次。在生成模型中，输入是由随机噪声转换成的深而窄的特征图，需要通过转置卷积实现上采样，从而生成 256 pixel×256 pixel 的彩色裂缝图像。之所以选择转置卷积，是因为相较于紧邻上采样、双样条采样、三次样条采样等手工选择采样方法，更需要一种能够自动优化的上采样方法，而且转置卷积已经在 DCGAN 中得到了良好的应用效果。

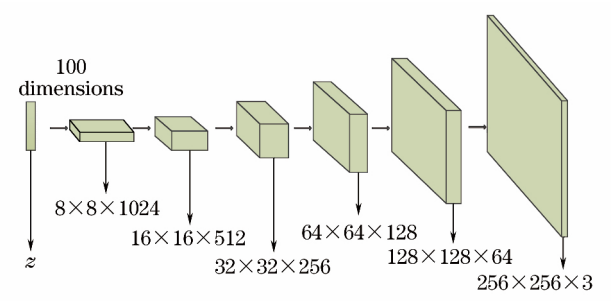

图 2 生成模型

Fig. 2 Generative model

之所以选择 5×5 的卷积核，是因为在深度学习中，常用的卷积核为 3×3、5×5、7×7，且卷积核越小，参数量越低，复杂度越小，但是，如果在桥梁裂缝检测中选择 3×3 大小的卷积核，不仅不利于提取裂缝的结构信息，还会受到噪声的影响。步幅设置为 2 是因为转置卷积的步幅定义了输出层的大小，且在“相同”的填充下，步幅为 2 时，输出特征的大小将是输入层的两倍。卷积核的数量与特征图大小的设置综合考虑了模型的复杂度与生成裂缝图像的真实程度。卷积层数目代表了模型的复杂度，层数越多精确度往往更高，但同样消耗的计算资源也更高。实验证实，上述设置可在模型复杂度最低的条件下，生成更真实的桥梁裂缝图像。

BCIGM 选择 SeLU 激活函数代替 Relu 激活函数和 Batch Normalization 的原因有：第一，SeLU 引入了自归一化的属性，使神经元激励值可以自动地收敛到零均值和单位方差，并且即使是存在噪声和扰动，通过许多层的前向传播后仍可收敛到零均值和单位方差。此外，对于不逼近单位方差的激励值，其方差存在上确界和下确界，因此梯度消失和梯度爆炸不可能出现，从而极大地提高了 BCIGM 的稳定性。

2.2.3 判别模型

判别模型的作用是通过特征提取判断输入样本是否是真实样本，由全卷积网络构成。具体过程如下：首先，输入 256 pixel×256 pixel×3 的样本图像，包括真实样本与生成样本两类；接着，经过 6 个卷积核大小为 5×5，步幅为 2，卷积核数目依次为 64，128，256，512，1024，2048 的卷积，再通过 1 个卷积核为 1×1 的卷积；最后，通过 Sigmoid 激活函数，映射出输入样本概率。加入 1×1 的卷积核是由于其可以在不改变特征图大小的情况下进行降维，减少参数的个数，从而降低计算时间。判别模型如图 3所示。

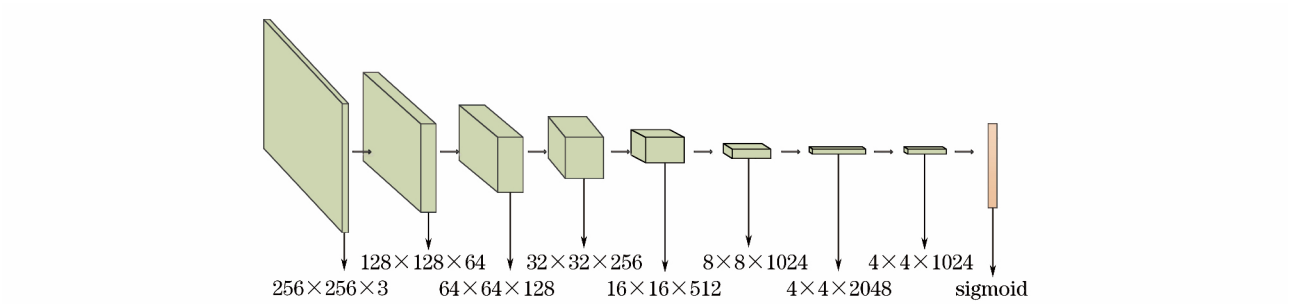

图 3 判别模型

Fig. 3 Discriminant model

3 基于语义分割的桥梁裂缝图像分割模型

图像语义分割[14-15]作为人工智能领域的重要分支，是计算机视觉中实现图像理解的重要一环，SegNet[16]、FCN[17]、DeepLab[18]等网络模型在多目标分类中已取得了较好的效果。众所周知，在一定条件下，网络的深度越深，提取到的特征越精确，检测效果也越好，但是在实际应用中，网络越深越容易出现梯度扩散的问题。2017 年，由 Huang 等[19]提出的密集卷积网络（DenseNet）解决了这个问题。因此，这里提出一种基于 DenseNet 的裂缝分割模型。

3.1 密集卷积网络

DenseNet 是一种具有密集连接的卷积神经网络。在该网络中，任何两层之间都有直接的连接，即网络每一层的输入都是前面所有层输出的并集，而该层所学习的特征图也会被直接传给其后面所有层作为输入。DenseNet 不仅可以高效利用特征图，还可以在网络很深时有效解决梯度消失问题。DenseNet 可表示为

$$X_l = H_l([X_0, X_1, \cdots, X_{l-1}]), \tag{2}$$

式中：l 为层数；X_l 代表 l 层的输出；$[X_0, X_1, \cdots, X_{l-1}]$表示将 0 到 $l-1$ 层的输出特征图按深度链接，即 Filter Concatenation。使用 Filter Concatenation 的好处在于防止了由层数增多带来的计算资源的爆炸性需求，从而扩展了网络的宽度和深度。$H_l(\cdot)$由 Batch Normalization、ReLU 激活函数和卷积操作组成。由于 Filter Concatenation 要求特征图 $X_0, X_1, \cdots, X_{l-1}$的尺寸相同，而池化操作会改变特征图尺寸且不可或缺，因此，提出 DenseBlock，使得 DenseBlock 内的特征图尺寸相同，每经过 DenseBlock 的一层，特征图数量增加 k，通过控制 k 可以控制网络的宽度，4 层 DenseBlock 的示意图如图 4 所示。

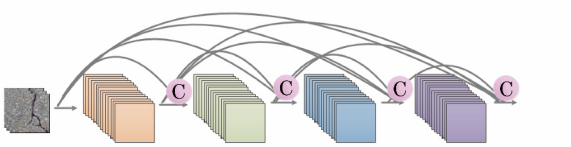

图 4 4 层 DenseBlock 的示意图

Fig. 4 Schematic of 4-layer DenseBlock

3.2 桥梁裂缝图像分割模型的构建

Jégou 等[20]提出用于语义分割的 FC-DenseNet103 模型，并在 CamVid 数据集中取得了满意的效果，但是当应用于复杂背景下的桥梁裂缝提取时，提取效果不佳，参数众多，训练模型时间较长。因此，针对 FC-DenseNet103 模型进行了改进，提出了适用于复杂背景的桥梁裂缝图像分割模型，简称 BCISM。

这里选择 FC-DenseNet103 模型作为基础模型，原因有以下几点：第一，桥梁裂缝为线性拓扑结构且图像背景纹理复杂，语义分割作为一种针对像素级别的精确分割，可以对每个像素进行预测，用于桥梁裂缝的检测有充足的理论支撑。第二，采用 DenseNet 的网络结构不仅能够提高模型的参数使用效率，而且到每个特征图都有短路径连接，从而可实现隐含的深层监督。此外，特殊的设计使所有层都可以很容易地访问到它们前面的所有层，这使得前面计算的特征图信息易被重用。第三，FC-DenseNet103 网络模型中 103 层的模型结构可以大幅提高精确度，并且有效避免了梯度消失，这是大部分语义分割模型无法实现的。

BCISM 一共由 74 个卷积层构成，网络模型具体包括由 DenseBlock 和 Transition Down 组成的下采样路径，由 DenseBlock 和 Transition Up 组成的上采样路径，以及 Softmax 函数。其中，DenseBlock 依次由 4，5，7，10，12，10，7，5，4 个层构成，每个层由 Batch Normalization、ReLU 激活函

数、3×3 卷积和 Dropout 构成。Dropout 是指在深度学习网络的训练过程中，对于神经网络单元，按照一定的概率将其暂时从网络中丢弃，以使每一个 batch 都在训练不同的网络，网络中 Dropout 设为 0.2。Transition Down 的作用是减少特征图的空间维度，由 Batch Normalization、ReLU 激活函数、1×1 卷积与 2×2 池化操作组成，其中 1×1 的卷积用于保存特征图的数量，2×2 的池化操作用于降低特征图的分辨率，以此来补偿由于网络层数大幅增加造成的特征图数量的线性增长。Transition Up 由一个转置卷积构成，作用是恢复输入图像的空间分辨率。转置卷积仅对最后一个 DenseBlock 的特征图使用，这是由于最后一个 DenseBlock 综合了所有之前 DenseBlock 的信息。Softmax 函数的作用是输出裂缝与非裂缝的概率。BCISM 的网络结构参数如表 1 所示。

表 1 BCISM 的网络结构参数

Table 1 Network structure parameters of BCISM

Name of layer	Size of kernel /(pixel×pixel)	Stride / pixel	Size of output feature map /(pixel×pixel)	Number of feature map
Inputlayer	—	—	256×256	3
Convolution	Convolution 5×5	1	256×256	48
DenseBlock	Convolution [3×3]×4	1	256×256	96
Transition Down	Convolution 1×1	1	256×256	96
	Max pooling 2×2	2	128×128	96
DenseBlock	Convolution [3×3]×5	1	128×128	156
Transition Down	Convolution 1×1	1	128×128	156
	Max pooling 2×2	2	64×64	156
DenseBlock	Convolution [3×3]×7	1	64×64	240
Transition Down	Convolution 1×1	1	64×64	240
	Max pooling 2×2	2	32×32	240
DenseBlock	Convolution [3×3]×10	1	32×32	360
Transition Down	Convolution 1×1	1	32×32	360
	Max pooling 2×2	2	216×16	360
DenseBlock	Convolution [3×3]×12	1	16×16	504
Transition Up	Deconvolution 3×3	2	32×32	504
DenseBlock	Convolution [3×3]×10	1	32×32	624
Transition Up	Deconvolution 3×3	2	64×64	624
DenseBlock	Convolution [3×3]×7	1	64×64	444
Transition Up	Deconvolution 3×3	2	128×128	444
DenseBlock	Convolution [3×3]×5	1	128×128	300
Transition Up	Deconvolution 3×3	2	256×256	300
DenseBlock	Convolution [3×3]×4	1	256×256	204
Convolution	Convolution 1×1	1	256×256	2
Softmax	—	—	—	—

3.3 高分辨率图像裂缝检测

在实际应用中桥梁裂缝图像分辨率都比较大，通常使用 512 pixel × 512 pixel，1024 pixel × 1024 pixel，甚至 2048 pixel × 2048 pixel 的图像。但是，深度学习网络中使用的输入图像分辨率一般都较小，如果将网络设计为高分辨率输入，数据量庞大会造成训练时间过久，甚至会由于网络模型太深、设计不当而造成模型无法收敛，从而无法得到理想的输出。因此，这里提出了针对高分辨率裂缝图像的检测算法。具体做法为：首先，用滑动窗口算法对高分辨率图像顺次进行裁剪，并对裁剪图像块按序标号；接着，将图像块依次输入 BCISM 进行裂缝检测并按原序依次输出；最后，将输出图像按原顺序拼接。512 pixel×512 pixel 的高分辨率裂缝图像检测示意图如图 5 所示，将原图顺次裁剪为(a)～(d)4 幅图像，依次输入 BCISM 后按原序输出为(a1)～(d1)，最后拼接为原图的检测结果。

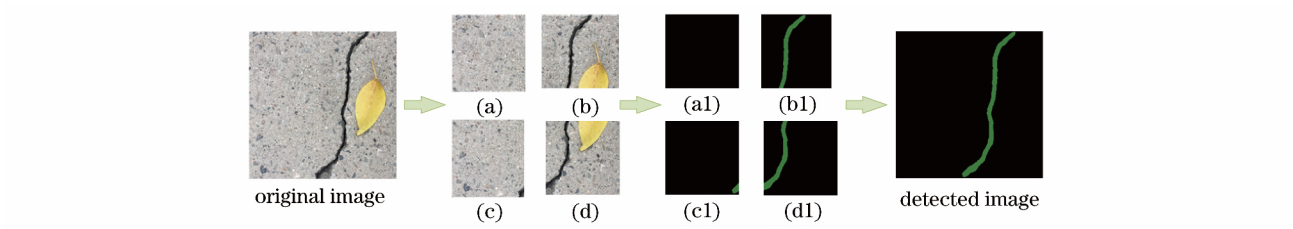

图 5 高分辨率图像检测示意图

Fig. 5 Schematic of detection of high-resolution image

4 实验结果与分析

4.1 数据集

实验共采集 1183 张桥梁裂缝图像，分为背景简单的图像、背景中含有障碍物的图像、背景中含有大面积污渍的图像 3 类。其中，背景简单的图像具体指裂缝图像中仅含有宽度较大的桥梁裂缝，其背景无任何干扰物；含有障碍物的图像具体指在裂缝图像的背景中包含各种各样的障碍物，实验中包括的障碍物类型有落叶、车道线、阴影、钉子、油漆、污渍等；背景中含有大面积污渍的图像具体指背景中存在大面积的污渍，其与第二类中含有污渍障碍物的区别为污渍所占面积比例的不同，此类中强调的是大面积。采集的裂缝图像原始分辨率为 2448 pixel×3264 pixel，为了便于在后续算法中使用，对原始图像进行预处理：首先，将采集来的图像短边缩放到 2048 pixel，裁出中央 2048 pixel×2048 pixel的区域；接着，将 2048 pixel×2048 pixel 的图像下采样为 1024 pixel×1024 pixel 的图像；再将1024 pixel×1024 pixel 的图像采用滑动窗口算法，不重叠裁剪为 16 幅 256 pixel×256 pixel 大小的图像；最后，通过第 2 节所述方法进行数据集扩充，最终不同类型图像的数量在数据集总量中的占比如表 2 所示。

表 2 不同类型图像的数量在数据集总量中的占比

Table 2 Proportion of number of different types of images in total dataset

Type of picture	Simple background	Background with obstacles	Background with large area of stains
Number of pictures	10449	13275	5710
Proportion /%	35.5	45.1	19.4

4.2 实验环境

算法程序基于主流的深度学习开源框架 TensorFlow，使用 Python 语言开发；实验的硬件环境为 Intel i7 处理器，NVIDIA GeForce GTX 1070 显卡；软件环境为 Ubuntu 16.04 LTS 操作系统。

4.3 对比实验

为了验证所提算法的有效性和准确性，分别设计了 5 组对比实验进行验证。

第 1 组实验用于验证 DCGAN 模型与 BCIGM 对于生成 256 pixel×256 pixel 的裂缝图像效果的影响。该实验包含 3 个小对比实验。实验 1 为 DCGAN 模型与 BCIGM 生成裂缝图像的可视化对比。实验 2 为 BCIGM 采用 SeLU 激活函数与采用 ReLU 激活函数生成裂缝图像的可视化对比。这 2 组实验选取前 4 张生成图像作为代表，N_{epoch} 表示 epoch 的数量。实验 3 为无 1×1 卷积核的 BCIGM 与有 1×1 卷积核的 BCIGM 训练网络时每个 batch 的使用时间对比，使用时间通过对每个 batch 的运行时间进行求和取平均值计算得到。

DCGAN 与 BCIGM 生成裂缝的可视化对比如图 6 所示，对应实验 1 的结果。ReLU 与 SeLU 生成裂缝的可视化对比如图 7 所示，对应为实验 2 的结果。表 3 为实验 3 的结果。通过观察实验 1 的 2 组图像可以看出，原 DCGAN 模型生成的裂缝图像网格现象严重，且学习到第 16 个 Epoch 时才初步出现条状裂缝特征，而所提出的 BCIGM 最后生成的裂缝图像清晰，基本无网格现象，与真实采集到的裂缝图像相似度极高，并且第 3 个 epoch 就出现了连接的线性裂缝特征。通过观察实验 2 可以看出，采用 ReLU 激活函数的 BCIGM 在第 3 个 epoch 时仅出现了不连续的黑色裂缝点，而采用 SeLU 激活函数的 BCIGM 出现了连续的裂缝雏形，除此之外，最后生成的裂缝图像也更加清晰，更符合实际场景下拍摄的裂缝图像，更重要的是提高了模型的稳定性，在测试期间没有出现任何一次模型崩塌现象。通过观察表 3 可以看出增加了 1×1 卷积核以后的 BCIGM 训练速度变快，虽然每个 batch 提升速度不多，但当训练数据集庞大，epoch 增加时，仍然可以极大地缩短计算时间。

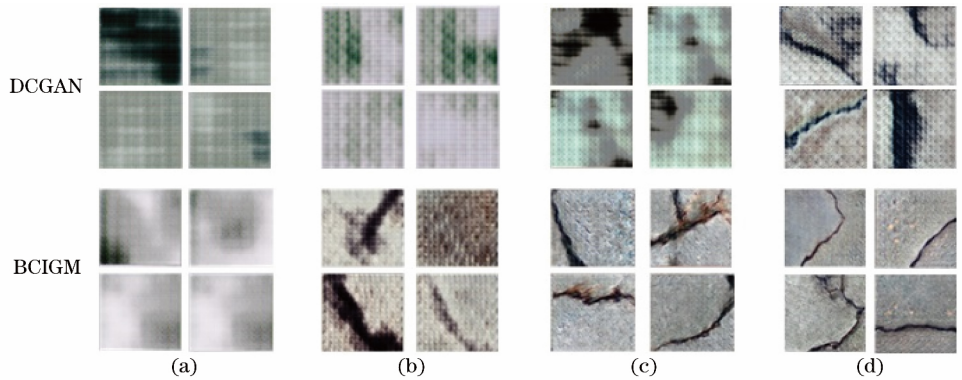

图 6 DCGAN 与 BCIGM 生成裂缝的可视化对比。(a) $N_{epoch}=01$;(b) $N_{epoch}=03$;(c) $N_{epoch}=16$;(d) $N_{epoch}=25$

Fig. 6 Visualization comparison of cracks generated by DCGAN and BCIGM. (a) $N_{epoch}=01$; (b) $N_{epoch}=03$; (c) $N_{epoch}=16$; (d) $N_{epoch}=25$

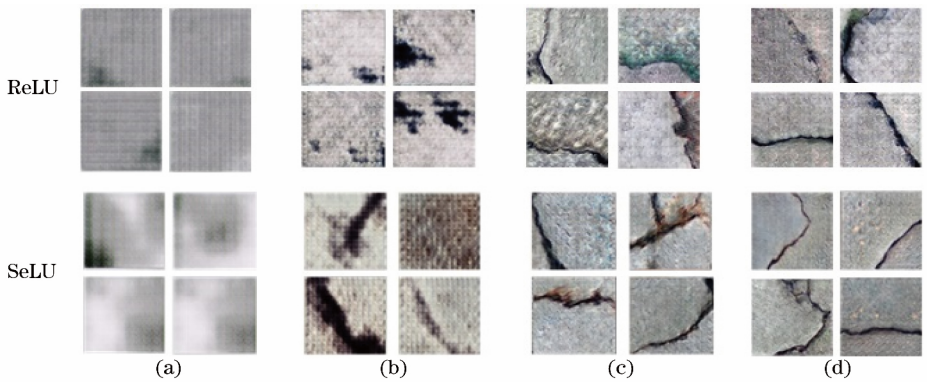

图 7 ReLU 与 SeLU 生成裂缝的可视化对比。(a) $N_{epoch}=01$;(b) $N_{epoch}=03$;(c) $N_{epoch}=16$;(d) $N_{epoch}=25$

Fig. 7 Visualization comparison of cracks generated by ReLU and SeLU. (a) $N_{epoch}=01$; (b) $N_{epoch}=03$; (c) $N_{epoch}=16$; (d) $N_{epoch}=25$

表 3 不同条件下 BCIGM 对训练速度的影响

Table 3 Influence of BCIGM on training speed under different conditions

Condition	Time /s
Without 1×1 convolution kernel	5.4801
With 1×1 convolution kernel	5.4312

第 2 组实验用于验证数据集扩增方法对于 BCISM 的影响。实验的具体操作步骤为:首先,直接使用人工采集的 1183 张桥梁裂缝图像在不经过数据集扩增方法的情况下训练 BCISM;接着,使用第 2 节所描述的数据集扩增方法进行扩增,使用扩增之后的数据集训练 BCISM;最后,从数据集的测试集中随机选取 156 张桥梁裂缝图像分别对训练好的 BCISM 进行测试。测试结果从 2 个角度进行评价,第 1 个角度为有无数据集扩增对实验结果的可视化对比,如图 8 所示。第 2 个角度采用量化的指标精确率 $P_{Precision}$ 与召回率 P_{Recall} 进行评价。这里,N_{TP} 代表被正确检测出来的裂缝区域像素的数量,N_{FP} 代表被误判为裂缝区域像素的数量,N_{FN} 代表属于裂缝区域的像素但是却没有被检测出来的像素的

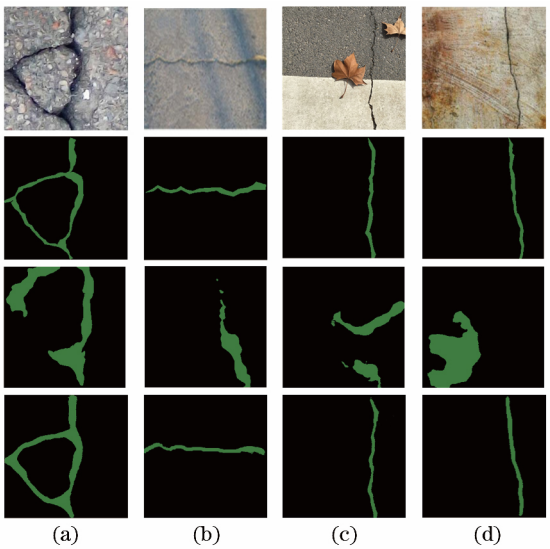

图 8 有无数据集扩增对实验结果的可视化对比。(a)原图;(b)标签;(c)无数据集扩增;(d)有数据集扩增

Fig. 8 Visualization comparison of experimental results with and without dataset amplification. (a) Original image; (b) label; (c) without dataset amplification; (d) with dataset amplification

数量，精确度 $P_{\text{Precision}}$ 与召回率 P_{Recall} 的具体计算公式为

$$P_{\text{Precision}}=\frac{N_{\text{TP}}}{N_{\text{TP}}+N_{\text{FP}}},\tag{3}$$

$$P_{\text{Recall}}=\frac{N_{\text{TP}}}{N_{\text{TP}}+N_{\text{FN}}}。\tag{4}$$

数据集扩增对实验结果的影响如表 4 所示。

由图 8 可知，利用未经过数据扩增的数据集训练的 BCISM 进行测试时，裂缝部分与原图中裂缝匹配度很低，检测效果较差。当裂缝背景相对简单时，未经数据集扩充模型的检测结果与标签图有相似之处，如图 8(a)列所示。但是当背景中存在障碍物或大面积污渍时，未经数据集扩充的模型的检测结果几乎与标签图像无任何相似度，如图 8(b)～(d)列所示。观察表 4 可知，第 1 组实验没有对采集所得的数据进行扩充，导致网络模型的训练样本严重不足，从而产生了欠拟合现象，其直接表现就是精确率与召回率极低，而利用数据集扩充之后的数据集训练的 BCISM 进行测试时，可以看到精确率与召回率显著提高，网络模型的欠拟合现象基本消除。综上所述，充足的训练样本对于模型的训练成败至关重要，这也证明了所提的数据集扩增方法是极其必要的。

表 4　数据集扩增对实验结果的影响

Table 4　Effect of dataset amplification on experimental results

Number of training samples	With or without dataset amplification	Number of verification samples	$P_{\text{Precision}}$ /%	P_{Recall} /%
1183	Without dataset amplification	156	13.5	17.9
29434	With dataset amplification	156	92.9	92.6

第 3 组实验用于验证所提出的 BCISM 与当前主流的语义分割模型的对比。本组实验选用了 SegNet、FCN、DeepLab、以及 FC-DenseNet56、FC-DenseNet67、FC-DenseNet103 模型作为 BCISM 的对比模型，对是否预训练、参数大小、精确率 P_{Precison}、召回率 P_{Recall}、模型精确率和召回率的加权平均 $P_{\text{F1\_Score}}$ 以及训练时每张图所用时间几方面进行对比。BCISM 与现有语义分割模型的对比如表 5 所示，其中，k 为增加的特征图数量，精确率 $P_{\text{Precision}}$ 与召回率 P_{Recall} 与第 4 章第 2 组对比实验所述一致，$P_{\text{F1\_Score}}$ 兼顾了精确率和召回率的查全和查准的作用。$P_{\text{F1\_Score}}$ 的具体计算公式为

$$P_{\text{F1\_Score}}=2\times\frac{P_{\text{Precision}}\times P_{\text{Recall}}}{P_{\text{Precision}}+P_{\text{Recall}}}。\tag{5}$$

由表 5 可知，BCISM 与 SegNet、FCN8、DeepLab 3 个模型相比，在没有预训练的情况下，参数量与时间均较低，且 $P_{\text{Precision}}$、P_{Recall}、$P_{\text{F1\_Score}}$ 均大幅提高。BCISM 与 FC-DenseNet56 相比，虽参数量与时间稍高，但 $P_{\text{Precision}}$、P_{Recall}、$P_{\text{F1\_Score}}$ 均提高 4%左右；BCISM 与 FC-DenseNet67 相比，参数量与时间均低，且 $P_{\text{Precision}}$、P_{Recall}、$P_{\text{F1\_Score}}$ 均提高；BCISM 与 FC-DenseNet103 相比，参数量不及三分之一且每张图的时间减少 0.1 s 左右，但是二者的 $P_{\text{Precision}}$、P_{Recall}、$P_{\text{F1\_Score}}$ 结果相近，均在 92%以上；综上所述，BCISM 在没有预训练的情况下，用较少的参数、较短的训练时间，达到了最精确的检测结果。

表 5　BCISM 与现有语义分割模型的对比

Table 5　Comparison of exiting semantic segmentation models and BCISM

Model	Pre-training	Parameter /M	$P_{\text{Precision}}$ /%	P_{Recall} /%	$P_{\text{F1\_Score}}$ /%	Time /s
SegNet	True	29.5	74.0	78.5	76.2	0.5823
FCN8	True	134.5	86.9	83.4	85.1	0.3739
DeepLab	True	37.3	82.6	80.9	81.7	0.9751
FC-DenseNet56 (k=12)	False	1.5	89.8	87.6	88.7	0.1685
FC-DenseNet67 (k=16)	False	3.5	89.0	88.8	88.9	0.2635
FC-DenseNet103 (k=16)	False	9.4	93.0	92.1	92.5	0.2795
BCISM (k=12)	False	2.8	92.9	92.6	92.8	0.1998

第 4 组实验用于验证所提算法与现有裂缝检测算法的裂缝检测效果对比。现有算法与所提算法的裂缝检测结果的对比如图 9 所示。

第 4 组对比实验选择两幅背景简单的裂缝图像，两幅含有障碍物的裂缝图像，一幅含有大面积污渍的裂缝图像作为裂缝图像的典型代表。选择阈值分割算法、Canny 算法作为传统裂缝检测算法代表，选择 NB-CNN 算法、随机结构森林算法作为深度学

习算法代表进行对比。通过观察实验结果，由图 9(c)列可见，通过阈值分割算法进行检测的结果会产生大量的噪声，并且当背景复杂时，会受到障碍物和大面积污渍的影响，导致检测效果更差。由图 9(d)列可见，Canny 算法的检测结果同样不理想，提取出的裂缝中非裂缝部分占据了极大的比例。由图 9(e)列可见，NB-CNN 算法可以检测到图像中所有的裂缝，但是仅可以标注裂缝所在位置，不能实现精确提取，而且把车道线、大片污渍等全部纳入裂缝提取框中。通过随机结构森林算法检测的图像，噪声量减少，但是当存在障碍物钉子与落叶时，会把钉子与部分落叶边缘错分为裂缝，而且当背景含有大面积污渍时，检测效果极差，如图 9(f)列所示。但是，由图 9(g)列可见，所提的复杂背景下基于图像处理的桥梁裂缝检测算法在以上几种情况下均取得理想效果，准确提取出了裂缝部分。因此可以证明：与现有算法相比，所提算法更适用于复杂背景下的桥梁裂缝检测。

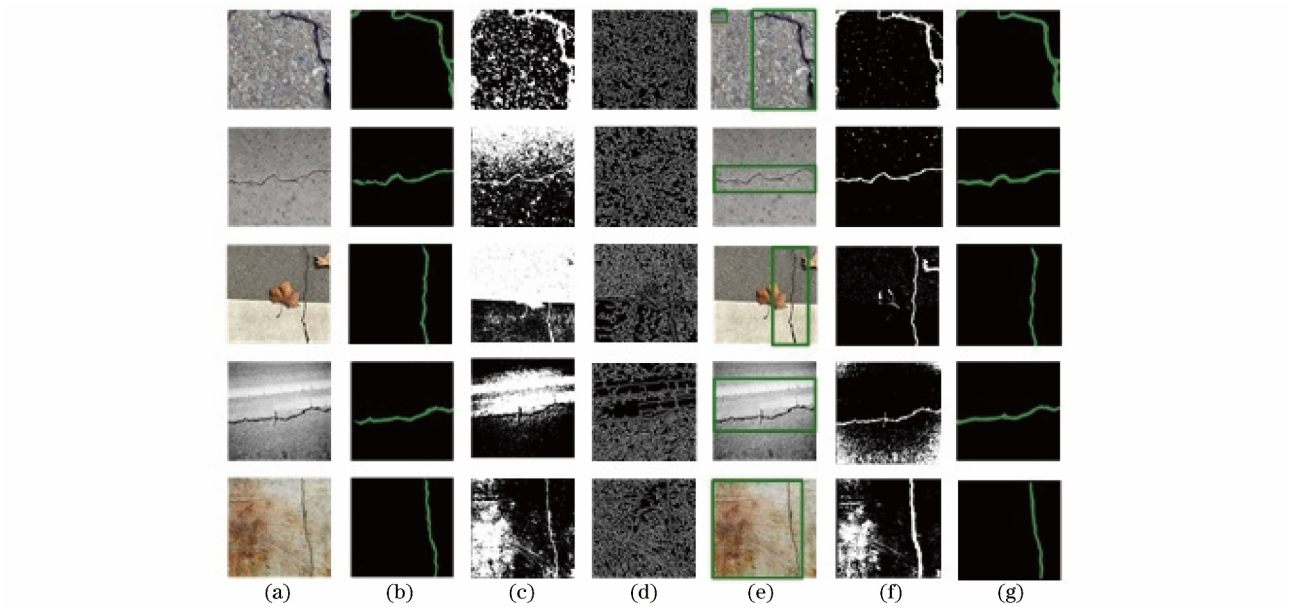

图 9 现有算法与所提算法的裂缝检测结果的对比。(a)原图；(b)标签；(c)阈值分割算法；(d) Canny 算法；(e) NB-CNN 算法；(f)随机结构森林算法；(g)所提算法

Fig. 9 Comparison of crack detection results between existing algorithms and proposed algorithm. (a) Original image; (b) label; (c) threshold segmentation algorithm; (d) Canny algorithm; (e) NB-CNN algorithm; (f) random structure forest algorithm; (g) proposed algorithm

第 5 组对比实验用于验证所提算法在背景简单的裂缝图像、背景含有障碍物的裂缝图像以及背景含有大面积污渍的裂缝图像 3 种场景中进行裂缝检测的有效性。利用所提算法对测试集中不同背景的桥梁裂缝图像进行了检测，部分裂缝检测结果如图 10所示。

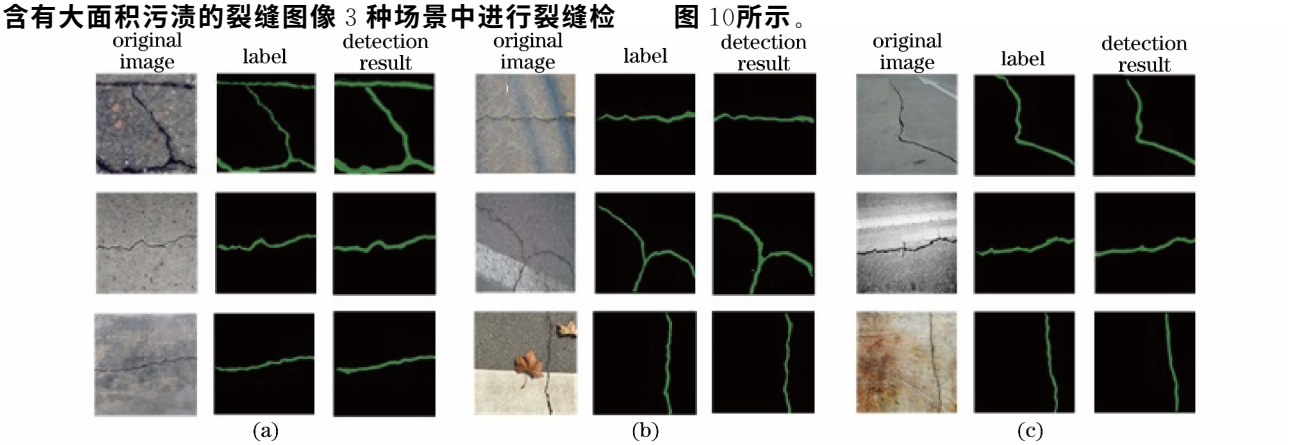

图 10 所提算法的部分裂缝检测结果。(a)场景 1；(b)场景 2；(c)场景 3

Fig. 10 Partial crack detection results by proposed algorithm. (a) Scene 1; (b) scene 2; (c) scene 3

为了证明所提算法适用于复杂背景的桥梁裂缝检测，第5组实验挑选了图10中的9幅原始图像进行说明。按照由上至下，由左及右的顺序对图10中的原始图像依次进行编号。第1幅、第2幅与第3幅为不同类型但背景均简单的裂缝图像，第4幅为背景中含有阴影的裂缝图像，第5幅为背景中含有车道线的裂缝图像，第6幅为背景中含有落叶与车道线的裂缝图像，第7幅为背景中含有油漆箭头与小块污渍的裂缝图像，第8幅为光照不均且含有车道线、钉子等障碍物的裂缝图像，第9幅为含有大面积污渍的裂缝图像。通过观察上图实验结果可以看出，所提算法不仅适用于背景简单的桥梁裂缝图像检测，而且适用于背景复杂的桥梁裂缝检测，可以证明所提算法的有效性。

5 结 论

提出了复杂背景下基于图像处理的桥梁裂缝检测算法。主要介绍了基于深度卷积生成式对抗网络的桥梁裂缝图像生成模型，提出了基于语义分割的桥梁裂缝图像分割模型，并实现了对高分辨率图像的裂缝提取。实验结果表明：数据集扩充有效缓解了由于数据不充足引起的欠拟合现象；与现有的语义分割模型相比，所提的桥梁裂缝图像分割模型参数量减少、训练时间缩短，但是检测效果提高；与现有桥梁裂缝检测算法相比，所提算法在复杂背景下有更理想的检测效果。

参 考 文 献

[1] Wang W Q. Development and expectation of bridge engineering technology [J]. Construction Technology, 2018, 47(6): 103-108.
王武勤. 桥梁工程技术发展与展望[J]. 施工技术, 2018, 47(6): 103-108.

[2] Qu L, Wang K R, Chen L L, *et al*. Fast road detection based on RGBD images and convolutional neural network[J]. Acta Optica Sinica, 2017, 37(10): 1010003.
曲磊，王康如，陈利利，等. 基于RGBD图像和卷积神经网络的快速道路检测[J]. 光学学报, 2017, 37(10): 1010003.

[3] National Bureau of Statistics of People's Republic of China. The statistics communique on national economy and social development of China 2017[N/OL]. China Information News, 2018-02-28(003). https://baijiahao.baidu.com/s?id=1593656271082733578&wfr=spider&for=pc.
中华人民共和国国家统计局. 中华人民共和国2017年国民经济和社会发展统计公报[N/OL]. 中国信息报, 2018-02-28(003). https://baijiahao.baidu.com/s?id=1593656271082733578&wfr=spider&for=pc.

[4] Deng X L, Tian S Z. Bridge deformation detection and data processing based on 3D laser scanning[J]. Laser & Optoelectronics Progress, 2018, 55(7): 071201.
邓晓隆，田石柱. 基于三维激光扫描的桥梁变形检测及数据处理[J]. 激光与光电子学进展, 2018, 55(7): 071201.

[5] Wang B, Wang X, Chen F, *et al*. Pavement crack recognition based on aerial image[J]. Acta Optica Sinica, 2017, 37(8): 0810004.
王博，王霞，陈飞，等. 航拍图像的路面裂缝识别[J]. 光学学报, 2017, 37(8): 0810004.

[6] Oh H, Garrick N W, Achenie L. Segmentation algorithm using iterative clipping for processing noisy pavement images[C]// Imaging Technologies: 2nd International Conference on Techniques And Application In Civil Engineering, May 25-30, 1997, Davos, Switzerland. Washington: TRB Publications, 1998: 138-147.

[7] Sun L, Xing J C, Xie L Q, *et al*. An adaptive threshold-based Canny algorithm for crack detection [J]. Microcomputer & Its Applications, 2017, 36(5): 35-37, 41.
孙亮，邢建春，谢立强，等. 基于自适应阈值Canny算法的裂缝检测方法研究[J]. 微型机与应用, 2017, 36(5): 35-37, 41.

[8] Talab A M A, Huang Z C, Xi F, *et al*. Detection crack in image using Otsu method and multiple filtering in image processing techniques[J]. Optik, 2016, 127(3): 1030-1033.

[9] Zhang L, Yang F, Zhang Y D, *et al*. Road crack detection using deep convolutional neural network[C]// IEEE International Conference on Image Processing, September 25-28, 2016, Phoenix, AZ, USA. New York: IEEE, 2016: 3708-3712.

[10] Chen F C, Jahanshahi M R. NB-CNN: Deep learning-based crack detection using convolutional neural network and naïve bayes data fusion[J]. IEEE Transactions on Industrial Electronics, 2018, 65(5): 4392-4400.

[11] Shi Y, Cui L M, Qi Z Q, *et al*. Automatic road crack detection using random structured forests[J]. IEEE Transactions on Intelligent Transportation

Systems, 2016, 17(12): 3434-3445.

[12] Radford A, Metz L, Chintala S. Unsupervised representation learning with deep convolutional generative adversarial networks[C]// International Conference on Learning Representations 2016 (ICLR 2016), May 2-4, 2016, San Juan, Puerto Rico. New York: Cornell University Library, 2016: 1511.06434.

[13] Klambauer G, Unterthiner T, Mayr A, *et al*. Self-Normalizing neural networks [C] // Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA. New York: Cornell University Library, 2016: 1706.02515.

[14] Li F X, Carreira J, Lebanon G, *et al*. Composite statistical inference for semantic segmentation[C]// 2013 IEEE Conference on Computer Vision and Pattern Recognition, June 23-28, 2013, Portland, OR, USA. New York: IEEE, 2013: 3302-3309.

[15] Guo C C, Yu F Q, Chen Y. Image semantic segmentation based on convolutional neural network feature and improved superpixel matching[J]. Laser & Optoelectronics Progress, 2018, 55(8): 081005.
郭呈呈, 于凤芹, 陈莹. 基于卷积神经网络特征和改进超像素匹配的图像语义分割[J]. 激光与光电子学进展, 2018, 55(8): 081005.

[16] Badrinarayanan V, Kendall A, Cipolla R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(12): 2481-2495.

[17] Shelhamer E, Long J, Darrell T. Fully convolutional networks for semantic segmentation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2017, 39(4): 640-651.

[18] Chen L C, Papandreou G, Kokkinos I, *et al*. DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 834-848.

[19] Huang G, Liu Z, Maaten L V D, *et al*. Densely connected convolutional networks [C] // IEEE Conference on Computer Vision and Pattern Recognition, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 2261-2269.

[20] Jégou S, Drozdzal M, Vazquez D, *et al*. The one hundred layers tiramisu: Fully convolutional DenseNets for semantic segmentation [C] // IEEE Conference on Computer Vision and Pattern Recognition Workshops, July 21-26, 2017, Honolulu, HI, USA. New York: IEEE, 2017: 1175-1183.