

基于改进的 PixelCNN++ 网络纹理合成方法

潘俊安¹ 段润涛² 丁要男¹ 张煜¹
PAN Junan DUAN Runtao DING Yaonan ZHANG Yu

摘要

纹理合成是从少量样本纹理中生成任意数量相似纹理的关键技术。早期纹理合成方法在合成阶段需依赖样本纹理图像,灵活性受限;后续发展的机器学习方法虽摆脱了合成阶段对样本的依赖,但存在非端到端设计的缺陷,且在复杂纹理场景下性能表现不佳。鉴于神经网络具备拟合任意非线性函数的能力且支持端到端学习,文章提出采用 PixelCNN++ 模型实现任意大小纹理图像的合成。该模型不仅可直接扩展至约束纹理合成与纹理修复任务,无需额外训练过程,有效降低了跨任务适配成本;同时,为解决纹理合成效率问题,进一步提出一种循环分配预测优化方法,以提升合成速度。实验表明,所提方法在纹理合成的灵活性、复杂场景适应性及效率方面均具有显著优势,为纹理生成领域提供了新的有效技术路径。

关键词

纹理合成; 纹理图像; PixelCNN++; 循环分配预测

doi: 10.3969/j.issn.1672-9528.2025.09.028

0 引言

纹理合成是通过从小样本图像的纹理中提取信息合成任意数量的相似纹理的过程。纹理合成在计算机视觉、计算机图形学和图像处理中发挥着重要作用。常见的有纹理修复^[1]和约束纹理合成^[2]。随着增强现实和虚拟现实应用的发展,快速高效的纹理合成变得更加重要。

约束纹理合成和纹理修复是基于纹理合成的进一步应用,也是基于原始图像来修复内部受损或缺失的纹理,需要确定缺失纹理的位置来进行纹理合成。

早期的纹理合成通常分为过程纹理合成(PTS)和样本纹理合成(TSFS)。PTS是通过物理生成过程的仿真直接在曲面上生成纹理,如毛发、云雾、木纹等,从而避免了纹理映射带来的失真。这种方法可以获得非常逼真的纹理,但对每一种新的纹理,却需要调整参数反复测试,非常不便,有的甚至无法得到有效的参数。缺点明显,通用性差,需要反复调整参数。TSFS无需重复调整参数。TSFS通常分为基于点的合成^[3]和基于块的合成^[4-7]。基于点的算法的缺点是速度慢,容易丢失纹理中的全局特征。基于块的算法可以解决基于点的合成算法的问题,选择足够的块也可以解决全局

特征缺失的问题。但是在贴图的接合处存在不连续的问题。上述纹理合成方法在整个合成阶段总是需要一个样本纹理。相比之下,通过学习获得的纹理取消了参数调整方法,显著提高了现有纹理合成方法的实用性。Gatys等人^[8]将CNN用于纹理合成^[9-10]。然而,无论是早期的方法还是最近的CNN、GAN和VAE都不能生成任意大小的纹理图像。Cai等人^[11]提出使用3个LSTM网络分别合成纹理的第一行、第一列和剩余像素。尽管这种方法可以合成任何大小的纹理图像,但LSTM所消耗的资源会随着合成图像大小的增加而增加。因此,本文提出PixelCNN++用于端到端任意大小的纹理合成。此外,本文方法可以直接扩展到基于原始图像的约束纹理合成和纹理修复,而无需额外的训练,这是GAN和VAE所没有的。为了解决PixelCNN++生成过程缓慢的问题,本文提出了一种基于循环分配预测的方法来提高纹理合成的速度。

1 相关工作

1.1 纹理合成

Guo等人^[12]提出了一种基于基因表达编程的水泥微观结构纹理合成方法。利用基因表达编程,分析并找到场中像素之间的关系,直接合成水泥微观结构纹理。该方法可以快速生成纹理图像,但不能搜索到所有的非线性关系。因此,通过这种方法合成的图像效果没有达到期望的结果^[13]。Dong等人^[2]使用支持向量机进行纹理合成。这种方法在合成接近规则或规则的纹理时特别有效。以上两

1. 江西机电职业技术学院 江西南昌 330013

2. 广州工商学院 广东广州 510800

[基金项目] 江西省教育厅科学技术研究项目“基于深度学习的混凝土裂缝检测技术与运用”(GJJ2406912)

种方法都是基于马尔可夫随机场元素之间的相关性原理进行纹理合成的,需要复杂的特征提取,有时无法很好地拟合复杂的纹理图像。

随着深度学习的发展,基于深度学习的纹理合成越来越受到关注。Fan 等人^[13]提出了一种基于前馈神经网络的水泥微观结构纹理合成方法。该方法的速度和效果优于基于基因表达编程的方法,但不能生成图像边缘像素信息。因此,实验结果仍然不理想。Ustyuzhaninov 等人^[14]提出使用前馈卷积网络代替 Gatys^[8]优化过程。尽管速度要快得多,但生成的样本缺乏多样性。在 Gatys 的基础上,Sendik 等人^[15]提出了基于 CNN 特征之间深度相关性的纹理合成。通过引入结构矩阵来表示深度特征之间的深度相关性,但其合成速度较慢。

为了从局部细节结构和整体结构两方面保证合成效果,谢莎莎^[16]提出了一种结合生成对抗网络(GAN)和图卷积网络(GCN)的非均匀纹理扩展合成模型(GAGCN)。GAN 学习图像块的内部分布,使合成图像具有丰富的局部细节。GCN 根据图像的统计特性学习像素之间的潜在相关性。为提升修复图像的纹理度,李海燕等人^[17]基于并行卷积残差连接修复网络,通过结合注意力机制,降低了组网损失,较经典算法,有效地提升了细节的修复能力。

1.2 纹理修复

纹理修复用于恢复检测区域中丢失的纹理。Efros 等人^[3]提出了一种约束纹理合成方法,该方法可以填充纹理上已知的空白区域。尽管这种方法取得了良好的效果,但在合成空白区域中的像素时,必须搜索整个纹理。Dong 等人^[2]将支持向量机应用于约束纹理合成和图像修复。但是需要复杂的特征提取。相反,本文方法是端到端的,不需要复杂的纹理搜索。

2 方法简介

本文方法框架如图 1 所示。首先训练 PixelCNN++;然后,通过 PixelCNN++ 合成新纹理,一次合成一个像素,直到合成所有像素以输出临时图像。

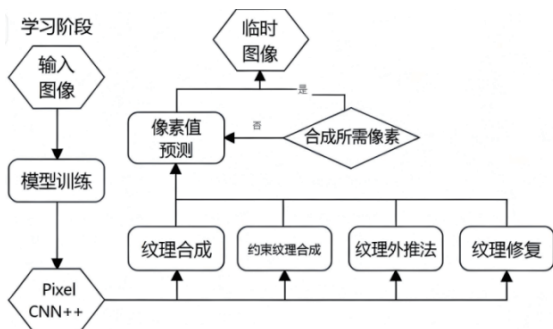


图 1 纹理合成过程

请注意,本文模型已经通过训练,可以直接用于纹理合成、约束纹理合成和纹理修复。

2.1 PixelCNN++

PixelCNN^[18]由 Google DeepMind 的 Oord 提出。PixelCNN 使用多个卷积层来保持空间信息,并使用掩码卷积代替池化层。虽然它比 PixelRNN 快,但效果不如 PixelRNN,而且由于使用了掩码卷积,PixelCNN 存在盲点问题。然后,Oord 提出了 Gated PixelCNN 的优化模型^[19]。该模型提高了性能,大大降低了计算成本。在此基础上,Salimans 等人^[20]提出了 PixelCNN++。PixelCNN++ 在网络的最后一层使用离散的逻辑混合似然,而非 256 种分类;使用整个像素作为条件,而不是 R/G/B 通道。

PixelCNN++ 的结构如图 2 所示。PixelCNN++ 由卷积网络组成,卷积网络可分为 3 个部分:卷积层、编码层和解码层。第一行的卷积核大小为 2×3。在第二行中,第一部分的卷积核是 1×3 和 2×1,其他卷积核是 2×2。在编码阶段,使用步长为 2 的卷积核来实现下采样操作。在解码阶段,使用步长为 2 的转置卷积来恢复到原始大小。

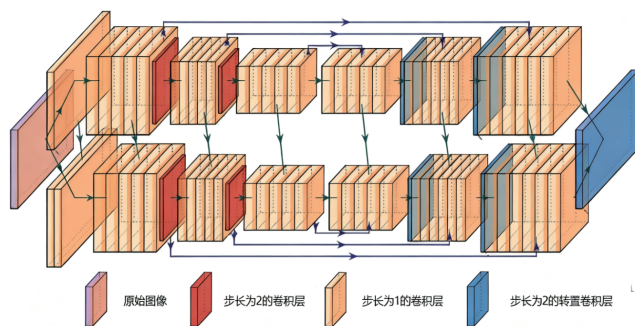


图 2 PixelCNN 的框架 ++

2.2 纹理合成与改进

图像 $U_{x \times y}$ 的生成过程为:将 $I_{x \times y}$ 输入 PixelCNN++ 并输出结果 p ;PixelCNN++ 将预测像素的 3 个通道输出到图像 $U_{i \times j}$,重复此操作,直到生成 $x \times y$ 大小的图像,最后输出图像 $U_{x \times y}$ 。

$$P = F(I_{x \times y}) \quad (1)$$

$$U_{i \times j} = G(p) \quad (2)$$

式中: $i=1,2,\dots,x$; $j=1,2,\dots,y$; $F()$ 表示 PixelCNN++; $I_{x \times y}$ 表示输入图像; p 表示模型的输出; $G()$ 表示采样和生成函数。

2.3 循环分配预测

通过此前测算可以看出,每个像素都需要放入模型中进行一次合成。这个过程非常耗时。使用 k_H 、 k_W 、 P 、 S 、 C_m 、 C_{out} 、 H_{out} 和 W_{out} 分别表示卷积核的行、列、填充大小、步长、输入通道和输出通道的数量、宽和高输出特性。即在具有偏差项的卷积神经网络的一层中的计算次数为:

$$H_{out}^1 = \frac{x - k_H + 2 \times P}{S} + 1 \quad (3)$$

$$W_{out}^1 = \frac{y - k_W + 2 \times P}{S} + 1 \quad (4)$$

$$m^1 = 2 \times k_H \times k_W \times C_m \times C_{out} \times H_{out}^1 \times W_{out}^1 \quad (5)$$

$$N^1 = x \times y \times m^1 \quad (6)$$

根据马尔可夫随机场，当两个像素之间的距离较大时，它们的相关性非常小。当输入图像的子区域为 $p \times q$ 时，具有偏差项的卷积神经网络中的计算次数为：

$$H_{out}^2 = \frac{x / y - k_H + 2 \times P}{S} + 1 \quad (7)$$

$$W_{out}^2 = \frac{y / q - k_W + 2 \times P}{S} + 1 \quad (8)$$

$$m^2 = 2 \times k_H \times k_W \times C_m \times C_{out} \times H_{out}^2 \times W_{out}^2 \quad (9)$$

$$N_2 = x \times y \times m^2 \quad (10)$$

从式(3~9)可以看出， N_1 的计算量小于 N_2 。当网络层数越深或生成图像的分辨率越大，计算的差异就会越明显。因此，考虑如何从图像中获得区域，以及在合成当前区域时如何生成下一个区域。关于如何获取区域，可以使用索引每次获取图像的行和列。在生成行和列之后，合成图像中的相应区域。因为它们是相同的记忆区域。为了解决如何合成下一区域的问题，将上一区域的合成像素信息添加到下一区域。有 3 种情况，基于前一个区域统称为 I 行或 I 列。这避免了生成图像的不连续性，并且不会出现明显的纹理分离。因为每个周期都会生成一个区域，所以这种方法被称为周期分配预测。该过程如图 3 所示，每次将子区域中的一个区域放入模型中以合成子区域的像素值。如果区域合成完成，取下一个子区域，否则继续合成该个子区域。直到图像的所有区域被组合。

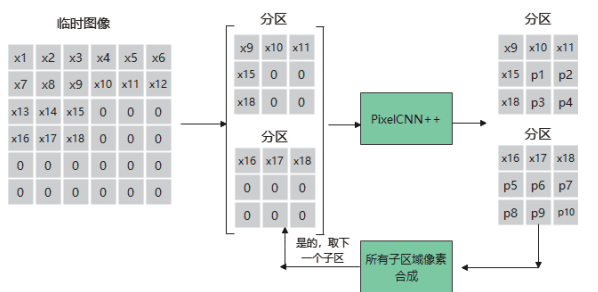
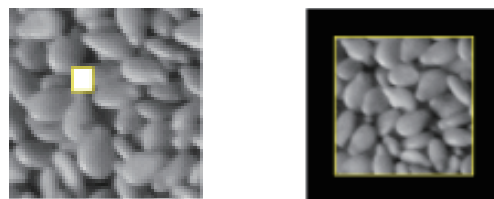


图 3 循环分配预测示意图

3 纹理合成的应用

如图 4 所示，在纹理合成的具体应用中，需要计算黄色区域的四个顶点坐标。本文提出了一种计算四顶点坐标的可行方法。首先，找到像素值等于 0（退化）或 255（缺失）的整个图像的坐标，并分别取第一维度和第二维度中的最大和最小坐标。此时，由顶点形成的区域是所需的黄色区域。对

于约束的纹理合成和纹理修复，只需要在黄色区域进行纹理合成。



(a) 约束的纹理合成 (b) 纹理修复

图 4 纹理合成的应用

4 实验与讨论

本文使用 PyTorch 来实现算法，在 Intel 3.8 GHz、128 GB RAM、NVIDIA GeForce RTX 2080Ti 和 Ubuntu 16.04 的服务器上进行了实验。纹理图像选自 Kylberg texture Dataset v.1.0 的 sesameeds1、canvas1、cushion1、scarf1 数据集和可见水泥数据集中标识号为 p35h4v1c300 的水泥数据。水泥数据的网站是：<http://visiblecement.nist.gov/>。本文模型合成的大小是 100 px×100 px。PixelCNN++ 的剩余块通道的数量为 64，并且批量大小为 64。基于循环分配预测，基于先前区域选择 20 行或 20 列。

图 5 是模型的纹理合成结果。其中，(a) 是输入的原始纹理图像；(b) 是通过原始生成方法合成的纹理图像；(c) 是通过本文方法合成的纹理图像。

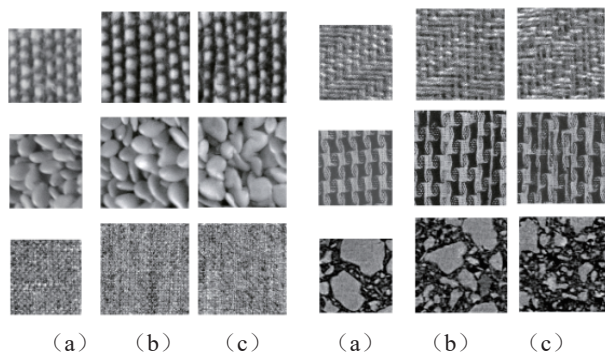


图 5 纹理合成实验结果

表 1 显示了原始生成方法和循环分配预测所消耗的时间比较。与原始方法相比，本文的方法在 GPU 上的时间减少了 2 倍多，在 CPU 上的时间缩短了近 15 倍。

表 1 原始方法和本文的方法所花费的时间比较

合成方法	CPU 时间 /min	GPU 时间 /min
原始方法	1 223.418 9	13.645 7
本文方法	82.211 0	5.931 0

图 6 (a) 和 (c) 是原始纹理图像；(b) 和 (d) 分别是纹理合成的结果。

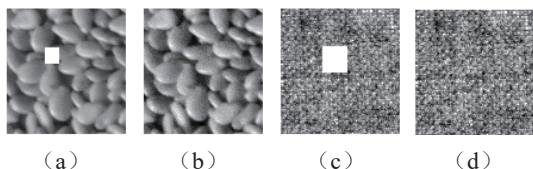


图6 基于 PixelCNN++ 的约束纹理合成

图7 (a) 和 (c) 是要修复的纹理图像; (b) 和 (d) 是修复后的纹理图像。可以发现, 本文方法在约束纹理合成和纹理修复方面也是有效的。

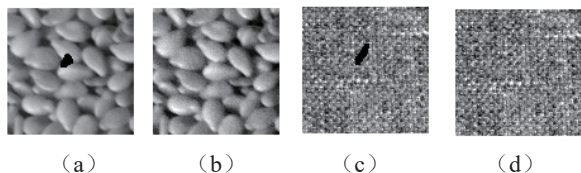


图7 基于 PixelCNN++ 的纹理修复

5 结论

本文使用 PixelCNN++ 来合成纹理图像。然而, 由于 PixelCNN++ 的合成速度较慢, 通过分析合成过程和马尔可夫随机场的相关性, 提出了一种基于循环分配预测的优化方法。实验结果表明, 在 CPU 上比原来的方法快 15 倍, 在 GPU 上比原来方法快 2 倍。

参考文献:

- [1] DAI D X, RIEMENSCHNEIDER H, SCHMITT G, et al. Example-based facade texture synthesis[C/OL]//2013 IEEE International Conference on Computer Vision. Piscataway:IEEE,2013[2025-05-12].<https://ieeexplore.ieee.org/document/6751242>.DOI:10.1109/ICCV.2013.136.
- [2] DONG X H, DONG J Y, SUN G M, et al.Learning-based texture synthesis and automatic inpainting using support vector machines[J]. IEEE transactions on industrial electronics, 2019, 66(6): 4777-4787.
- [3] EFROS A A, LEUNG T K. Texture synthesis by non-parametric sampling[C/OL]// Proceedings of the Seventh IEEE International Conference on Computer Vision. Piscataway: IEEE, 2002[2025-05-23].<https://ieeexplore.ieee.org/document/790383>. DOI:10.1109/ICCV.1999.790383.
- [4] EFROS A A, FREEMAN W T. Image quilting for texture synthesis and transfer[C]//SIGGRAPH'01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques. NewYork: ACM, 2001:341-346.
- [5] LIANG L, LIU C, XU Y Q, et al. Real-time texture synthesis by patch-based sampling[J]. ACM transactions on graphics (TOG), 2001, 20(3):127-150.
- [6] XIE Y M, PAN Z H, ZHANG D G,et al.A new kind of texture synthesis method based on best match[J].Applied mechanics & materials, 2014, 575:585-588.
- [7] REN Y, ROMANO Y, ELAD M .Example-based image synthesis via randomized patch-matching[J]. IEEE transactions on image processing, 2018,27(1):220 - 235.
- [8] GATYS L A, ECKER A S, BETHGE M .Texture synthesis using convolutional neural networks[EB/OL]. (2015-11-06) [2025-05-13].<https://doi.org/10.48550/arXiv.1505.07376>.
- [9] GATYS L A, ECKER A S, BETHGE M .A neural algorithm of artistic style[J/OL].Journal of vision, 2015[2025-06-11]. https://www.researchgate.net/publication/281312423_A_Neural_Algorithm_of_Artistic_Style.DOI:10.1167/16.12.326.
- [10] GATYS L A, ECKER A S, BETHGE M .Image style transfer using convolutional neural networks[C/OL]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).NewYork:ACM,2016[2025-03-23]. <https://ieeexplore.ieee.org/document/7780634>.DOI:10.1109/CVPR.2016.265.
- [11] CAI X X, SONG B, FANG Z Q .Exemplar based regular texture synthesis using LSTM[J].Pattern recognition letters, 2019,128:226-230.
- [12] GUO J F, ZHANG N, WANG L,et al.A fast texture synthesis using gene expression programming[C/OL]// 2016 3rd International Conference on Informative and Cybernetics for Computational Social Systems (ICCSS). Piscataway:IEEE, 2016[2025-05-23].<https://ieeexplore.ieee.org/document/7586435>.DOI:10.1109/ICCSS.2016.7586435.
- [13] FAN J W, WANG L, XIAO C,et al.Cement texture synthesis based on feedforward neural network[C/OL]//2019 International Conference on Security, Pattern Analysis, and Cybernetics.Piscataway:IEEE,2019[2025-03-13].<https://ieeexplore.ieee.org/document/9085129>.DOI:10.1109/SPAC49953.2019.244103.
- [14] USTYUZHANINOV I, BRENDDEL W, GATYS L A,et al.Texture synthesis using shallow convolutional networks with random filters[EB/OL]. (2016-05-31)[2025-04-14]. <https://doi.org/10.48550/arXiv.1606.00021>.
- [15] SENDIK O, COHEN-OR D. Deep correlations for texture synthesis[J]. ACM transactions on graphics (ToG), 2017, 36(5): 1-15.
- [16] 谢莎莎. 基于生成对抗网络与图卷积网络的纹理合成 [D]. 昆明: 云南大学, 2022.
- [17] 李海燕, 晁艳静, 李海江, 等. 并行生成卷积网络图像修复算法 [J]. 国防科技大学学报, 2023,45(3):127-135.

PECS-Net：一种融合多尺度局部 – 全局特征与双重注意力机制的断裂面识别算法

钟 玉^{1,2} 谢晓尧^{1,2} 刘 嵩^{1,2*} 刘建成³ 王 冲³ 刘旭斌^{1,2}
ZHONG Yu XIE Xiaoyao LIU Song LIU Jiancheng WANG Chong LIU Xubin

摘 要

在人类历史的长河中，积累了许多珍贵的文物。随着时间的推移和各种因素的影响，这些文物不可避免地遭受了损坏。修复受损的文物不仅有助于保护文化遗产，还对弘扬和传承历史文化也起着重要作用。对此，文章提出了一种基于深度学习的模型 PECS-Net，用于识别文物残片的断裂面和原始表面。该模型以改进的 PointNet++ 网络为基础，修改原有的特征提取方法，提取多尺度的局部 - 全局特征，同时结合双重注意力机制增强特征表示。测试结果表明，PECS-Net 模型在 FragTag3D 数据集和 3D Puzzles 数据集上分别实现了 94.241% 和 76.859% 的断裂面识别准确率，平均交并比(MIoU)得分分别为 0.874 和 0.623。与 Baseline 相比，PECS-Net 在上述数据集上将断裂面识别准确率提高了 19.529% 和 15.022%，MIoU 得分分别提高了 0.277 和 0.201。

关键词

PointNet++；多尺度局部特征；全局特征；注意力机制；断裂面识别；碎片拼接

doi: 10.3969/j.issn.1672-9528.2025.09.029

0 引言

文物的保护与修复具有极为重要的意义。传统上，受损文物的修复依赖专业人员手工拼接碎片。这种方法耗时费力，严重依赖专家的经验，同时在修复过程中可能会对文物造成二次损害。

近年来，随着计算机技术的飞速发展，计算机视觉领域涌现出大量人才和创新技术，使得具备强大计算能力的计算机能够在碎片匹配与拼接任务中替代人工^[1-3]。在利用计算机修复破损文物的过程中，根据碎片厚度的特征，可将其分为非薄壁碎片与薄壁碎片两类^[4]。对于非薄壁碎片，可以从断裂表面提取特征，并利用这些特征进行点对点匹配，以完成碎片的拼接。因此，准确高效地区分断裂表面与原始表面，成为实现非薄壁碎片自动拼接的关键步骤之一。

为了克服这一挑战，本文提出了一种新的网络——PECS-Net，该网络能够识别非薄壁文物碎片的断裂表面，为文物的

1. 贵州师范大学贵州省先进计算全省重点实验室
贵州贵阳 550001
2. 贵州师范大学网络空间安全学院 贵州贵阳 550001
3. 四川省文物考古研究院 四川成都 610093

[18] OORD A V D, KALCHBRENNER N, KAVUKCUOGLU K. Pixel recurrent neural networks[C]//ICML'16: Proceedings of the 33rd International Conference on International Conference on Machine Learning. New York: ACM, 2016: 1747 - 1756.

[19] OORD A V D, KALCHBRENNER N, VINYALS O, et al. Conditional image generation with PixelCNN decoders[EB/OL]. (2016-06-16)[2025-02-22]. <https://www.semanticscholar.org/paper/Conditional-Image-Generation-with-PixelCNN-Decoders-Oord-Kalchbrenner/0936352b78a52bc5d2b5e3f042>. DOI:10.48550/arXiv.1606.05328.

[20] SALIMANS T, KARPATHY A, CHEN X, et al. PixelCNN++: improving the PixelCNN with discretized logistic mixture

likelihood and other modifications[EB/OL]. (2017-01-19)[2025-03-22]. <https://doi.org/10.48550/arXiv.1701.05517>.

【作者简介】

潘俊安(1997—)，男，江西南昌人，硕士，教师、工程师，研究方向：图像处理、深度学习。

段润涛(1998—)，男，河南平顶山人，硕士，教师，研究方向：图像处理。

丁要男(1996—)，男，江西上饶人，硕士，教师，研究方向：图像处理、深度学习。

张煜(1996—)，女，江西南昌人，硕士，教师，研究方向：计算机视觉、深度学习。

(收稿日期：2025-03-24 修回日期：2025-09-03)