

文章编号: 2095-2163(2023)01-0091-09

中图分类号: TP183

文献标志码: A

基于 LBP 等价模式和注意力机制的图像修复算法

王浩明

(上海理工大学 光电信息与计算机工程学院, 上海 200093)

摘要: 为了解决目前主流的图像修复算法在面对内容复杂的图像修复时容易产生伪影与模糊的问题。本文提出了一种两阶段图像修复算法。在第一阶段的特征学习网络中,本文采用了局部二值模式(Local Binary Pattern, LBP)的等价模式获取更高质量的结构信息,用于指导第二阶段的修复;为了找到相似度更高的 patches,保证修复结果在整体和局部的一致性,本文在第二阶段修复网络的生成器中插入新的注意力层,同时在已知区域和生成区域中寻找相似 patches,以对缺失部分进行更好的填充。实验证明,本文提出的算法很好地改善了面对复杂图像时出现的问题,修复结果视觉效果更好,性能评价指标值更高。

关键词: 图像修复; 局部二值模式的等价模式; 注意力

Image inpainting algorithm

based on LBP uniform pattern and attention mechanism

WANG Haoming

(School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China)

[Abstract] In order to solve the problem that the current mainstream image inpainting algorithms tend to produce artifacts and blurring in the inpainting results when facing images with complex content, a new two-stage image inpainting algorithm is proposed in this paper. In the first stage--the feature learning network, in order to obtain structural information with higher quality to guide the inpainting of the second stage, the Uniform Pattern of Local Binary Pattern is adopted in this paper. In order to find patches with higher similarity and strengthen the global and local consistency of the inpainting results, this paper inserts a new attention layer into the generator of the inpainting network in the second stage to find similar patches in both the known and generated regions to better fill the missing parts. The experimental part also proves that the proposed algorithm can effectively improve the problems of facing complex images, and compared with other mainstream algorithms, the inpainting results obtained by the proposed algorithm have better visual effects and higher value in the performance evaluation metrics.

[Key words] image inpainting; uniform pattern of local binary pattern; attention

0 引言

图像修复是指运用技术手段对图像中缺失破损的区域进行重建的过程。这一技术被广泛应用于文物和老照片的修复、医学图像和卫星遥感图像的去噪、影视特效以及军事安全等多个领域。

目前图像修复算法中常采用两阶段网络结构:先在第一阶段网络中粗略地预测缺失的结构,再用预测的结构信息指导第二阶段网络进行更精细的修复,这是一个由粗到细的过程。Huang 等人^[1]在第一阶段使用一个简单的膨胀卷积网络,在第二阶段整合上下文注意力层来完成最终的修复;Liu 等人^[2]

提出的两阶段网络都是基于 U-Net 结构,为了获得更好的空间一致性,在第二阶段还额外引入了连贯语义注意力层;Ren 等人^[3]设计了 StructureFlow,在第二阶段保留边缘的平滑图像来训练结构重建器,在第二阶段使用带有外观流的纹理生成器生成图像细节;Nazeri 等人^[4]还提出了一个由边缘生成器紧接一个图像修复网络组成的 EdgeConnect 模型。

尽管两阶段网络模型已经取得一定成效,但是在面对内容复杂的图像时,得到的修复结果仍然容易产生伪影与模糊,使得最终修复结果的整体与局部的一致性都有所下降。造成这一结果的根本原因在于第一阶段得到的结构信息质量不高,不足以为

基金项目: 国家自然科学基金(61773083);上海市浦江人才计划(2019PJC073)。

作者简介: 王浩明(1997-),男,硕士研究生,主要研究方向:图像修复。

收稿日期: 2022-04-07

第二阶段提供良好的指导,且在第二阶段中不能使用最适合的 patches 来对缺失区域进行进一步填充,对两阶段网络的不同设计将产生不同的修复效果。

基于这一思路,本文在 Wu 等人^[5]的工作的启发下提出了一种新的两阶段网络,更好地解决面对复杂图像会出现伪影与模糊的问题。

1 算法整体概述

本文算法的整体流程如图 1 所示。在生成器 G_1 、 G_2 中,每一层上面的数字表示输入 $256 \times 256 \times 3$ 的图像时卷积层/反卷积层得到的每层特征图的尺寸,每一层下面的数字则表示卷积层/反卷积层每层的输入通道数。

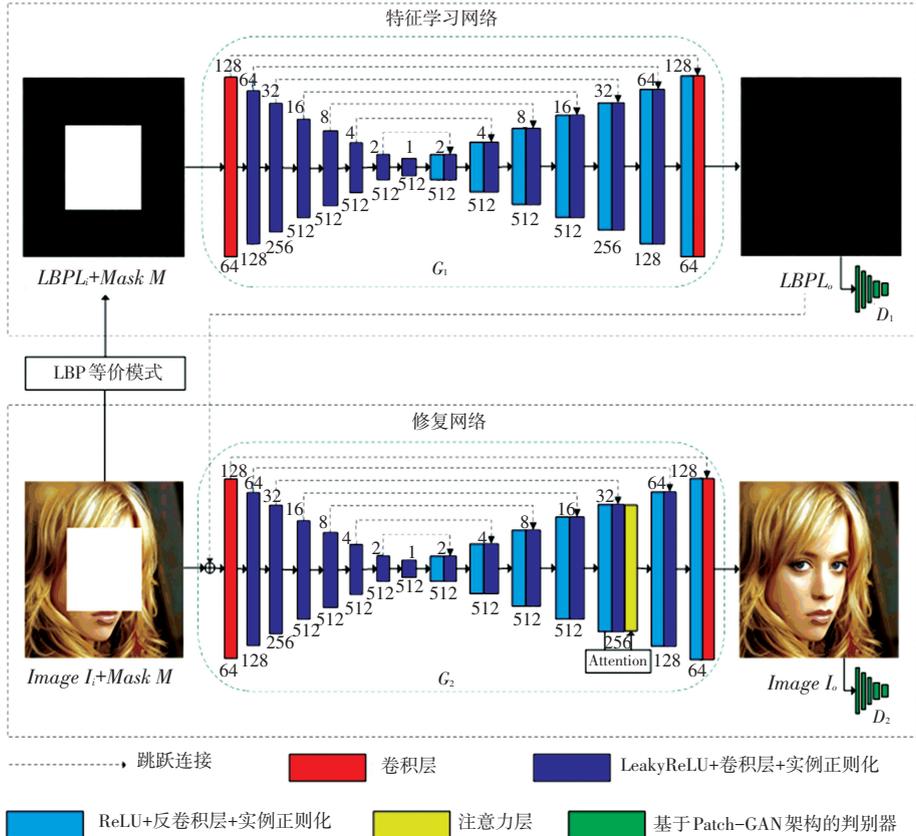


图 1 算法整体流程图

Fig. 1 Overall flow chart of the algorithm

1.1 特征学习网络流程

第一阶段的特征学习网络用于预测出完整的 LBP 结构信息,为第二阶段修复网络进一步的精细修复提供指导,是一种粗修复。

特征学习网络的输入由 I_i 的 LBP 特征图 L_i 以及合成 I_i 对应的 Mask M 组成。

其中, I_i 是由完整图片与 Mask M 融合生成的破损图片, LBP 特征的提取是通过 LBP 等价模式完成的;输入的 L_i 和 M 经过生成器 G_1 与判别器 D_1 的不断博弈,最终由 G_1 预测出完整的 LBP 特征图 L_o , 作为第二阶段修复网络输入的一部分。

1.2 修复网络流程

第二阶段的修复网络用于完成最终精细的修

复。修复网络的生成器 G_2 是以第一阶段的最终输出 L_o 以及 I_i 与其对应的 Mask M 作为输入的,通过 G_2 与修复网络的判别器 D_2 的联合训练,输出最终的修复结果 I_o 。

2 特征学习网络

特征学习网络主要由 LBP 等价模式模块、生成器 G_1 和判别器 D_1 组成。

2.1 LBP 等价模式

在许多两阶段图像修复网络中,都需要使用在第一阶段网络中捕获的结构信息为第二阶段网络提供指导,因此捕获到的结构信息质量将直接影响后续图像修复的质量。目前许多两阶段图像修复网络中使

用的结构信息,通常涉及许多参数,如预滤波强度和边缘响应阈值等,并且其最佳设置也因图像而异。

相比之下,LBP 特征涉及参数少,并且在纹理较细的区域,包含了更丰富的结构信息。此外,相比其他特征需要不断调整参数设置,来获取最佳效果,LBP 特征更加通用简单,因此本文选用 LBP 特征作为第一阶段的结构信息。在此基础上,本文使用了 LBP 的等价模式,通过设置半径 $R = 1$, 采样点 $P = 8$, 实现 LBP 算子模式种类数量的进一步减少。实验证明,本文采用的 LBP 等价模式对于最终的修复效果起到了很好的引导作用。

2.1.1 原始 LBP

局部二值模式(Local Binary Pattern,LBP)是一种用于描述图像局部纹理特征的算子,具有涉及参数少,包含丰富结构信息的优点^[6]。

原始 LBP 的计算方式示意如图 2 所示。以图像上任一像素点(如图 2 中像素值为 5 的黄色像素点)为中心,对其取一个 3×3 的邻域,依次将邻域中的其余 8 个点与中心像素点进行像素值的比较,比中心像素点像素值大的置 1,反之置 0;从左上角(如图 2 中原像素值为 3 的像素点)开始,顺时针依次将这些 0 和 1 连成一个八位二进制序列(图 2 中为 01100101);最后,将序列转换为十进制,便得到了这一像素点的 LBP 值;依次计算图像上的每一个像素点的 LBP 值,便可得到整张图像的 LBP 特征图。

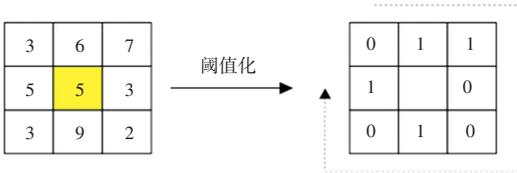


图 2 原始 LBP 计算方式示意图

Fig. 2 Schematic diagram of the original LBP calculation method

2.1.2 圆形 LBP 算法

为了增强 LBP 适应不同尺寸图片的能力,圆形 LBP 算法被提出^[7]。具体来说是通过使用半径为 R , 采样点总个数为 P 的圆形邻域代替正方形邻域,如图 3 所示。从左到右依次为:半径为 1,采样点数为 8;半径为 2,采样点数为 8;半径为 2,采样点数为 16 的示意图。

圆形 LBP 算法首先需要计算出除了中心像素点以外其余各个采样点的坐标,根据坐标找到对应位置的像素值,依旧按照前面的原始 LBP 计算方式计算出中心像素对应的 LBP 值,第 p 个采样点的坐

标 (x_p, y_p) 计算公式如式(1)所示。

$$\begin{cases} x_p = x_c + R \cos\left(\frac{2\pi p}{P}\right) \\ y_p = y_c - R \sin\left(\frac{2\pi p}{P}\right) \end{cases} \quad (1)$$

其中, (x_c, y_c) 为中心像素点坐标; R 表示半径; P 表示采样点总个数。

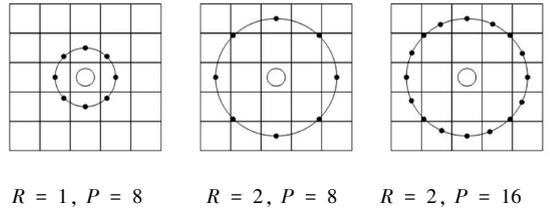


图 3 圆形 LBP 示意图

Fig. 3 Schematic diagram of Circular LBP

2.1.3 LBP 等价模式

由圆形 LBP 算法可以看出, P 个采样点对应着 2^P 种模式,这种指数对应关系,使得 P 即使变动不大,二值模式种类的数量也会发生巨大变化,这对于纹理的提取、识别和分类都是不利的。所以需要使用等价模式来减少 LBP 算子的模式种类的数量,实现对圆形 LBP 算法的进一步优化^[7]。

当某个 LBP 值所对应的二进制数从 0 到 1 或从 1 到 0 最多有两次跳变时,该 LBP 值所对应的二进制就定义为一个等价模式类,其余的都被称为混合模式类。

LBP 等价模式使二值模式的种类数量大大减少,由原来的 2^P 种减少为 $P(P - 1) + 1$ 种,而不会丢失任何信息。

2.2 生成器 G_1 概述

为了获得更精细的图像修复结果,本文的特征学习网络的生成器 G_1 保留了 Shift-Net 中生成器的 U-Net 架构,其完整的网络结构见表 1。Conv、DeConv、Cat、LReLU、IN 分别代表卷积层、反卷积层、连接层、 $\alpha = 0.2$ 的 LeakyReLU 以及实例正则化(InstanceNorm)。每个括号里的 4 个数字都分别代表滤波器数量(即通道数)、卷积核尺寸、步长以及 padding 数目。如:Conv(128, 4, 2, 1),括号里的 4 个数字分别表示滤波器数量为 128,卷积核尺寸为 4×4 ,步长为 2,padding 为 1。

G_1 的编码器部分(层 1-层 8)除了第一层是卷积层外,每层都是由 Leaky ReLU、卷积以及 InstanceNorm 层叠加而成,编码阶段各卷积层的输入通道数依次为 64,128,256,512,512,512,512,512。

表1 G_1 完整结构
Tab. 1 The complete structure of G_1

层数	每层详情
1	Conv(64, 4, 2, 1);
2	LReLU; Conv(128, 4, 2, 1); IN;
3	LReLU; Conv(256, 4, 2, 1); IN;
4	LReLU; Conv(512, 4, 2, 1); IN;
5	LReLU; Conv(512, 4, 2, 1); IN;
6	LReLU; Conv(512, 4, 2, 1); IN;
7	LReLU; Conv(512, 4, 2, 1); IN;
8	LReLU; Conv(512, 4, 2, 1); ReLU; DeConv(512, 4, 2, 1); IN;
9	Cat(Layer 8, Layer 7); ReLU; DeConv(512, 4, 2, 1); IN;
10	Cat(Layer 9, Layer 6); ReLU; DeConv(512, 4, 2, 1); IN;
11	Cat(Layer 10, Layer 5); ReLU; DeConv(512, 4, 2, 1); IN;
12	Cat(Layer 11, Layer 4); ReLU; DeConv(256, 4, 2, 1); IN;
13	Cat(Layer 12, Layer 3); ReLU; DeConv(128, 4, 2, 1); IN;
14	Cat(Layer 13, Layer 2); ReLU; DeConv(64, 4, 2, 1); IN;
15	Cat(Layer 14, Layer 1); ReLU; DeConv(3, 4, 2, 1); Tanh;

G_1 的解码器部分(层8-层15)都是由 ReLU、反卷积层以及实例正则化叠加而成。解码阶段各反卷积层输入通道数依次为 512、512、512、512、256、128、64、3。

2.3 判别器 D_1 概述

对于特征学习网络的判别器 D_1 , 本文采用 PatchGAN 架构, 见表2。 D_1 与 G_1 的编码器部分具有相似的设计模式, 但只有5个卷积层。除了第一层和第五层, 每一层的卷积层后都紧接一个实例正则化, 除了第一层, 每一层都有一个 $\alpha = 0.2$ 的 LeakyReLU 层。卷积层的卷积核尺寸都为 4×4 、步长都为2, padding 都为1, 各卷积层的输入通道数依次为 64、128、256、512、1。此外, 最后一层使用了 Sigmoid 作为激活函数。

表2 D_1 完整结构

Tab. 2 The complete structure of D_1

层数	每层详情
1	Conv(64, 4, 2, 1);
2	LReLU; Conv(128, 4, 2, 1); IN;
3	LReLU; Conv(256, 4, 2, 1); IN;
4	LReLU; Conv(512, 4, 2, 1); IN;
5	LReLU; Conv(1, 4, 2, 1); Sigmoid;

3 修复网络

修复网络主要由生成器 G_2 和判别器 D_2 组成。在生成器 G_2 中, 除了引入了新的注意层, G_2 与 G_1 、 D_2 与 D_1 都分别采用了完全相同的结构。

为了使修复结果同时在全局和局部上具备更好的一致性, 在修复网络的生成器中加入新的注意层。现有的使用注意力机制的算法都默认缺失区域与已知区域具有相同的结构特征, 仅在已知区域中搜寻相似 patches 来对缺失区域进行填充, 可能会遗漏掉更高质量的 patches。本文的注意层将搜索范围扩大到已知区域以及修复过程中生成的区域, 以得到更高质量的相似 patches。

(1) 相似性评价指标。对应于扩大的搜索范围, 本文使用两种相似性指标来评价 patches 间的相似性, 包括生成区域中的 patches 之间的余弦内相似性 $S_{j,k}$ 以及生成区域与已知区域的 patches 之间的余弦内相似性 $\bar{S}_{j,k}$ 。 P 中的 patches P_j 与 P_k 的余弦内相似性, 公式(2); P_j 与在 \bar{P} 中的 patches \bar{P}_k 的余弦内相似性, 公式(3):

$$S_{j,k} = \left\langle \frac{P_j}{\|P_j\|}, \frac{P_k}{\|P_k\|} \right\rangle, \quad P_k \in P \quad (2)$$

$$\bar{S}_{j,k} = \left\langle \frac{P_j}{\|P_j\|}, \frac{\bar{P}_k}{\|\bar{P}_k\|} \right\rangle, \quad \bar{P}_k \in \bar{P} \quad (3)$$

其中, P 代表从生成器 G_2 第13层的特征图中的生成区域中提取出的所有 1×1 的 patches 组成的集合, \bar{P} 则代表从生成器 G_2 第13层的特征图中的已知区域中提取出的所有 1×1 的 patches 组成的集合。

通过计算特征图中所有的 $S_{j,k}$ 和 $\bar{S}_{j,k}$, 便可以从

P 和 \bar{P} 中找到与 P_j 最相似的 patches。

(2) 相似性搜索。本文通过卷积的方式计算出所有的 $S_{j,k}$ 和 $\bar{S}_{j,k}$, 并根据数值大小分别选出 P 和 \bar{P} 中的 top - K patches, 因为使用更多的 patches 来更新缺失的区域时收效甚微, 还会增加计算成本, 所以本文最终设置 $K = 2$ 。

(3) 更新 patches。为了更好的确保修复结果的全局一致性, 本文采用非局部均值策略更新每个 $P_j \in P$, 更新的具体计算公式如式(4)、式(5)所示:

$$P_j^* = \sum_{k \in N} \frac{\exp(S_{j,k})}{Z_j} P_k + \sum_{k \in \bar{N}} \frac{\exp(\bar{S}_{j,k})}{Z_j} \bar{P}_k \quad (4)$$

$$Z_j = \sum_{k \in N} \exp(S_{j,k}) + \sum_{k \in \bar{N}} \exp(\bar{S}_{j,k}) \quad (5)$$

其中, P_j^* 代表更新后的 P_j , $N = \{n_1, n_2\}$ 和 $\bar{N} = \{\bar{n}_1, \bar{n}_2\}$, 分别记录 P 和 \bar{P} 中最相似的两个 patches 的索引, Z_j 则表示归一化因子。

此外, 将注意层安排在第 13 层是出于对计算资源和性能的综合考量: 进入解码器阶段, 越深的层得到的特征图的尺寸越大, 需要的计算资源需求也相应增加; 但是当层数较小, 提取的可用 patches 数量不足, 无法获得更相似的 patches 来填充破损区域, 最终的修复性能也会因此下降。因此, 为了在计算资源和性能之间实现平衡, 参照 Shift-Net 的设置, 本文最终选择在生成器的第 13 层插入新的注意层, 此时特征图的尺寸为 32×32 。

4 损失函数设计

4.1 特征学习网络损失函数设计

同主流的图像修复算法一样, 特征学习网络整体的损失函数 L_{LEARN} 是多个损失函数的加权和。 L_{LEARN} 表示最终预测到的 LBP 特征与 Ground Truth 的 LBP 特征的差距, 用于评估经过特征学习网络预测后得到的 LBP 特征的质量, 其具体计算方式为

$$L_{LEARN} = \omega_{rec1} L_{rec1} + \omega_{adv1} L_{adv1} \quad (6)$$

其中, L_{rec1} 表示重建损失; L_{adv1} 表示对抗损失; ω_{rec1} , ω_{adv1} 分别代表 L_{rec1} 和 L_{adv1} 各自的权重。

重建损失 L_{rec1} 选用了 L_1 损失, 具体计算方式为

$$L_{rec1} = \|L_o - L_g\|_1 \quad (7)$$

其中, L_o 表示预测出的 LBP 特征图, L_g 则表示 Ground Truth 的 LBP 特征图。

对抗损失 L_{adv1} 的具体计算方式为

$$L_{adv1} = \min_{G_1} \max_{D_1} E_{L_g} [\log D_1(L_g)] + E_{L_i} [\log(1 - D_1(G_1(L_i, M)))] \quad (8)$$

其中, M 表示二值掩码, L_i 表示从输入的破损

图片 I_i 中提取的 LBP 特征。

4.2 修复网络损失函数设计

修复网络的整体损失函数 L_{Inp} 在特征学习网络的整体损失函数 L_{LEARN} 的基础上额外引入了感知损失 L_{per} 和风格损失 L_{style} , 具体计算方式为

$$L_{Inp} = \omega_{rec2} L_{rec2} + \omega_{adv2} L_{adv2} + \omega_{per} L_{per} + \omega_{style} L_{style} \quad (9)$$

其中, ω_{per} , ω_{style} 为感知损失 L_{per} 和风格损失 L_{style} 对应的权重, ω_{rec2} , ω_{adv2} 为重建损失 L_{rec2} 和对抗损失 L_{adv2} 对应的权重。

感知损失 L_{per} 的计算方式为

$$L_{per} = \sum_{l \in C} \|\varphi_l(I_o) - \varphi_l(I_g)\|_1 \quad (10)$$

其中, I_o 表示最终修复得到的图片; I_g 表示 Ground Truth 图片; φ_l 表示对应于 ImageNet 预训练的 VGG-16 网络的第 l 层的激活图, 集合 C 包括了 conv2_1、conv3_1、conv4_1 层的层索引。

本文的风格损失函数 L_{style} 具体计算方式为

$$L_{style} = \sum_{l \in C} \|\mathbf{G}^{\varphi_l}(I_o) - \mathbf{G}^{\varphi_l}(I_g)\|_1 \quad (11)$$

其中, \mathbf{G}^{φ_l} 是给定大小为 $C_i \times H_i \times W_i$ 的特征图后, 由与感知损失相同的激活图 φ_l 构造的 $C_i \times C_i$ 的 Gram 矩阵。

5 实验

5.1 实验条件

5.1.1 数据集

本文实验共使用 4 种数据集, 即 CelebA-HQ、Places 和 Paris StreetView 以及 NVIDIA 提供的不规则 Mask 数据集。本文使用 CelebA-HQ 数据集原有的训练集用于训练, 原有的测试集对半分为本文的测试集与验证集, 即训练集 28 000 张, 新的测试集和验证集各 1 000 张; 本文从 Places 数据集中随机选择 10 种场景, 并将其原始验证集对半分成本文的新的验证集和测试集, 即训练集是来自 10 种场景的 5 万张图片, 验证集和测试集都分别是来自 10 种场景的 500 张图片; 本文从 Paris StreetView 数据集原始训练集中选择 100 张图像作为验证集, 使用剩下的 14 800 张图像来构成训练集, 测试集不变; 训练时采用从不规则 Mask 数据集中随机选取的策略。除了不规则 Mask, 本文还使用了中心 Mask, 中心 Mask 的尺寸统一采用 128×128 (单位为像素)。此外, 输入网络的图片都会预处理为 256×256 的大小。

5.1.2 训练环境

本文的实验模型主要是基于 Pytorch 1.6.0 和 Tensorflow 1.8.0 框架搭建的, 使用一台操作系统为

Ubuntu 的服务器进行训练。服务器的显卡型号为 RTX 3090, 显存大小为 24 G, 处理器类型为 i9-10900k、运行内存大小为 64 GB, 硬盘大小为 2 T。编程语言使用 python 3.6。

5.1.3 训练过程

首先在特征学习网络中训练生成器 G_1 和判别器 D_1 , 将 G_1 得到的结果连接到修复网络中, 同时对 G_1 、 G_2 和 D_2 进行训练。

5.1.4 部分参数设置

batchsize 设置为 8, 模型使用 Adam 优化器对参

数进行优化, 其中 $\beta_1 = 0, \beta_2 = 0.9$, 生成器网络的学习率设为 10^{-4} , 判别器的学习率为 10^{-5} 来对网络微调, 直至收敛为止, 最多训练 30 个 epoch。

5.2 定性比较

为了直观地验证本文提出算法的有效性及其优越性, 将本文的算法与当下流行的几种算法在 Places 数据集上进行了定性比较, 结果如图 4、图 5 所示。图 4、图 5 中的输入图片分别使用了中心 Mask 以及不规则 Mask, 且图 5 输入图片中的缺失比例自上而下不断扩大。

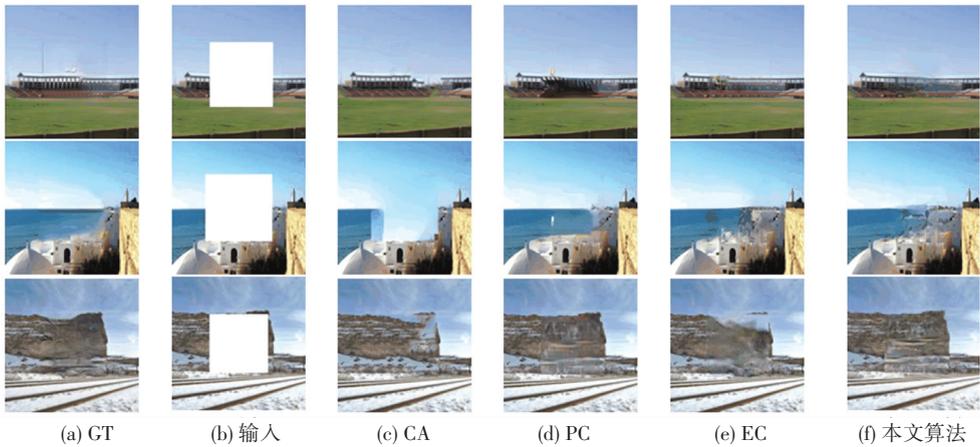


图 4 使用中心 Mask 的 Places 数据集上的修复结果对比图

Fig. 4 Comparison of each algorithm on Places dataset with central mask

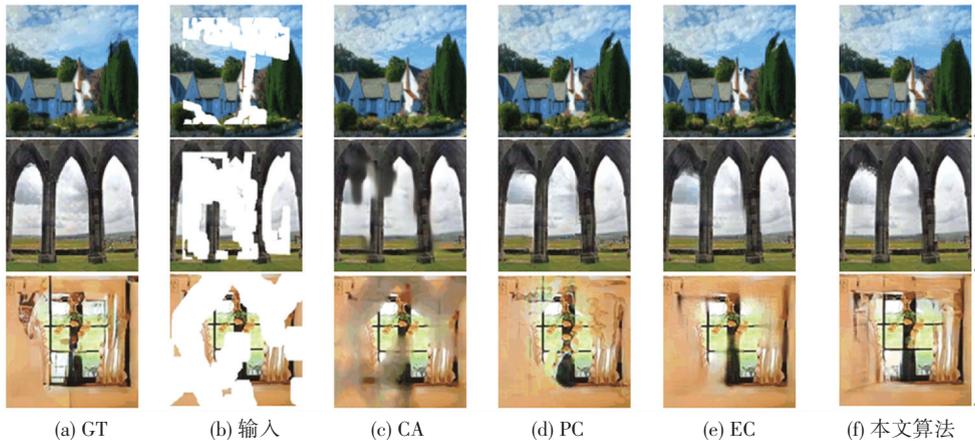


图 5 使用不规则 Mask 的 Places 数据集上的修复结果对比图

Fig. 5 Comparison of each algorithm on Places dataset with irregular mask

由图 4 可以看出, 使用了中心 Mask 的 Places 图片, CA 出现了严重的语义不符: 海面处出现了蓝色方框; EC 也出现了明显的掩膜边界: 山后的天空出现了明显的正方形边界; 本文算法的结果也在海面部分出现了模糊, 只有 PC 的结果细腻自然, 具备更多的纹理细节。

由图 5 可以看出, 面对使用不规则 Mask 的

Places 图片: 在缺失比例较小 ($0\% \sim 30\%$) 时, 所有算法均取得了较好的修复效果, 与 GT 图片相比, 结果也很逼真。但是当缺失比例逐渐扩大 ($\geq 30\%$), 除本文算法外的所有算法依次出现了模糊, 其中 CA 的模糊最为严重, PC 和 EC 均丢失了细节, 本文算法所有图片依旧产生了自然真实的修复效果。

图 4 从视觉上定性证明本文算法在使用了中心

Mask 的 Places 数据集上,性能未能达到最佳;图 5 则从视觉上定性证明了在使用了不规则 Mask 的 Places 数据集上,本文的算法产生了最好的修复效果。

5.3 定量比较

通过使用性能指标来对图像修复算法的性能进行客观的定量比较。本文采用图像修复中最常用的性能评价指标:峰值信噪比(Peak Signal to Noise Ratio, PSNR)和结构相似指数(Structural Similarity index, SSIM)。两种指标的值的的大小都是与算法的修复性能呈正相关的,PSNR 的单位为 dB,SSIM 的范围为 0~1。

Contextual Attention (CA)、Partial Convolutions (PC)、Edge Connect (EC) 以及本文算法在中心

Mask 的 CelebA-HQ 数据集和 Paris StreetView 数据集上得到的性能评价指标如图 6、图 7 所示。可以看出,在中心 Mask 的 CelebA-HQ 数据集上,本文算法的 PSNR 和 SSIM 值分别为 28.19 dB 以及 0.924;在中心 Mask 的 Paris StreetView 数据集上,本文算法的 PSNR 和 SSIM 值分别为 26.35 dB 以及 0.868;因此,无论在哪个数据集上,本文算法的 PSNR 和 SSIM 值都比当前流行的其他算法高,尤其是在 CelebA-HQ 中,本文算法的 PSNR 值比 CA 高出 4.95 dB,SSIM 值比 CA 高出 0.038,定量证明了本文算法与其他主流算法相比,在中心 Mask 的 CelebA-HQ 和 Paris StreetView 数据集上具有更好的修复性能,即从数值上反映了采用的 LBP 等价模式和新的注意力层对于最终修复效果的有效性。

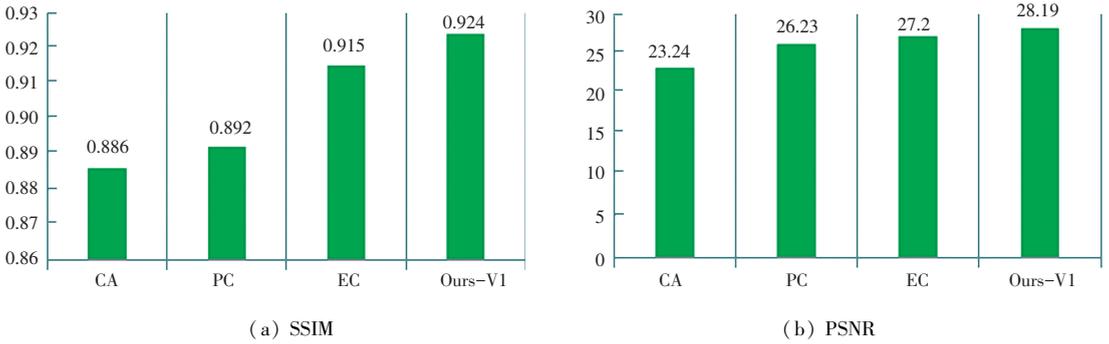


图 6 各算法在中心 Mask 的 CelebA-HQ 数据集上的 SSIM 与 PSNR 值对比图

Fig. 6 Comparison of SSIM and PSNR values obtained by each algorithm on CelebA-HQ dataset with central Mask

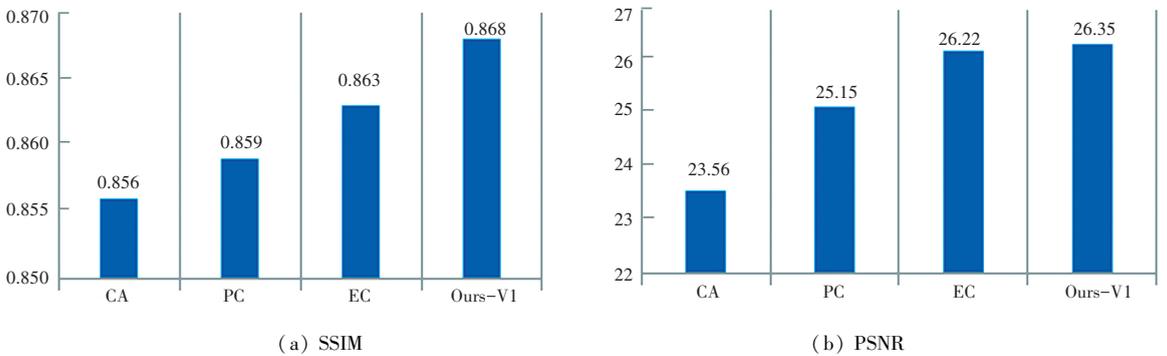


图 7 各算法在中心 Mask 的 Paris StreetView 数据集上的 SSIM 与 PSNR 值对比图

Fig. 7 Comparison of SSIM and PSNR values obtained by each algorithm on Paris StreetView dataset with central Mask

5.4 消融实验

为了验证本文提出的 LBP 等价模式和注意层对于图像修复任务的有效性,分别对这两部分依次进行消融实验。

5.4.1 LBP 等价模式

通过对 LBP 等价模式进行消融实验,得到了图 8 所示的结果。

由图 8(c)、图 8(g)对比可看出,未使用 LBP 等价模式得到的修复结果出现了明显的扭曲与细节丢失,眉毛与眼睛的细节都未能很好地还原;而使用 LBP 等价模式得到的修复结果则很好地还原了人脸的细节,产生了合理自然的结果,与 Ground Truth 图像也更接近,两图之间的对比很好地从视觉上定性证明了 LBP 等价模式的有效性。

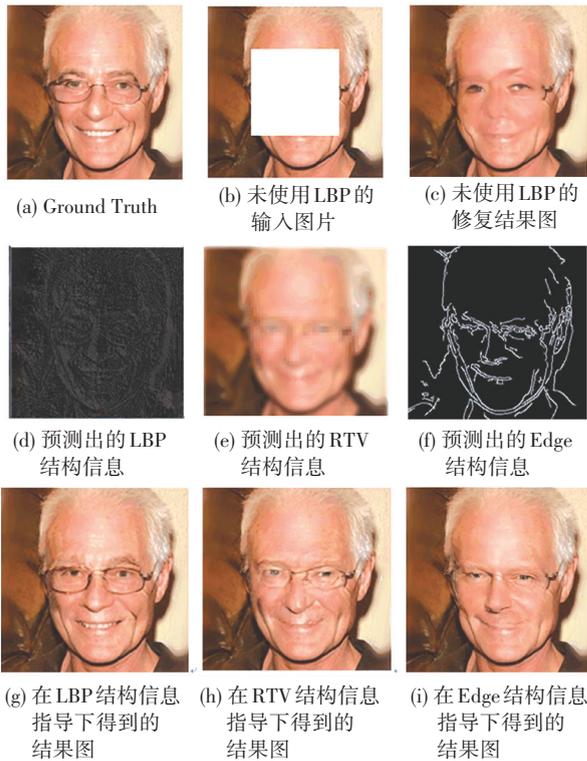


图8 LBP等价模式消融实验结果定性对比图

Fig. 8 Comparison of qualitative results of LBP uniform pattern ablation study

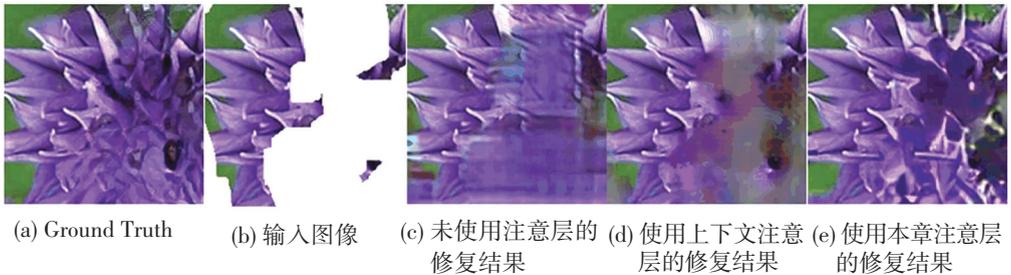


图9 注意层消融实验定性结果比较图

Fig. 9 Comparison of qualitative results of attention layer ablation study

由图9(c)、图9(e)对比可直观看出,加入上下文注意层后,尽管修复效果获得了一定的提升,去除了部分模糊,还原了部分细节;将图9(d)与图9(e)进行对比可以看出,使用本文提出的注意层能更好地还原结构信息,得到的修复的结果也更为清晰,从视觉上定性证明了本文提出的注意层相比其他注意层的优越性。

除了上述的定性比较,本文还使用了不规则Mask的Places数据集做了注意层的定量比较,最终结果见表3。

由表3的数据对比可看出,使用本文注意层的算法比不使用任何注意层的方法,PSNR值高出了4.919 dB,SSIM值高出了0.129,这一提升定量地验

由图8(d)、图8(e)、图8(f)对比可看出:结构重建器预测的RTV结构信息过于平滑,遗漏了大量的结构细节,边缘生成器预测的边缘结构信息包含不连续点,只有LBP等价模式预测的结果包含了更完整的结构信息。

由图8(g)、图8(h)、图8(i)对比可以看出,最终的修复结果也是经过LBP特征引导的最为清晰准确,很好地证明了LBP等价模式在提供结构信息方面相比其他结构特征提取器的优越性。

5.4.2 注意层

为了验证本文提出的注意层的有效性以及优越性,将不使用任何注意层、使用上下文注意层以及使用本文新的注意层得到的修复结果进行比较,比较的结果如图9所示。

由图9(c)、图9(e)对比可直观看出,使用与不使用本文提出的注意层最终得到的修复结果之间差距明显:不使用本文提出的注意层得到的结果十分模糊,花朵的大部分细节也已经丢失;而使用本文提出的注意层得到的结果则很好地还原了花朵缺失部分的细节,得到的结果也更加细腻自然,从视觉上定性证明了本文提出的注意层的有效性。

证了本文提出的注意层的有效性;使用本文注意层的算法比使用上下文注意层的方法,在PSNR上高出4.439 dB,在SSIM上高出0.067,定量地验证了本文提出的新的注意层的优越性。

表3 注意层消融实验结果定量比较表

Tab. 3 Comparison of quantitative results of attention layer ablation study

不同注意层	W/O Ours	CA	With Ours
SSIM	0.721	0.783	0.850
PSNR	21.181	21.661	26.100

6 结束语

为了更好地解决面对复杂图像时会出现伪影与模糊的问题,本文提出新的两阶段网络,在第一阶段

的特征学习网络中使用的 LBP 等价模式, 获得了更高质量的结构信息, 为第二阶段的修复提供了更好的指导。并且在第二阶段修复网络的生成器中插入的新的注意层同时在已知区域和生成区域中寻找更高质量的相似 patches, 使得最终的修复结果具备更好的全局和局部一致性。通过与其他算法进行主观的定性比较以及客观的定量比较, 证明本文所提出的修复算法更好地解决了内容复杂的图像出现的伪影、模糊、不连贯的问题, 能生成一致性更强、更加细腻真实的修复结果。但是, 在面对具有大面积不规则缺失的图像时, 本文算法出现了明显的性能下降, 后续工作将针对这一不足开展。

参考文献

[1] YU J, LIN Z, YANG J, et al. Generative image inpainting with contextual attention[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2018: 5505–5514.

[2] LIU H, JIANG B, XIAO Y, et al. Coherent semantic attention for image inpainting[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 4170–4179.

[3] REN Y, YU X, ZHANG R, et al. Structureflow: Image inpainting via structure-aware appearance flow[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 181–190.

[4] NAZARI K, NG E, JOSEPH T, et al. Edgeconnect: Generative image inpainting with adversarial edge learning[J]. arXiv preprint arXiv:1901.00212, 2019.

[5] WU H, ZHOU J, LI Y. Deep generative model for image inpainting with local binary pattern learning and spatial attention[J]. IEEE Transactions on multimedia, 2021.

[6] OJALA T, PIETIKÄINEN M, HARWOOD D. A comparative study of texture measures with classification based on featured distributions[J]. Pattern recognition, 1996, 29(1): 51–59.

[7] OJALA T, PIETIKAINEN M, MAENPAA T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns[J]. IEEE Transactions on pattern analysis and machine intelligence, 2002, 24(7): 971–987.

(上接第 90 页)

代替 Local BA 提高位姿优化的速度和精度。

在 KITTI 和 New Tsukuba 数据集上的实验结果表明, 本文改进后的 SLAM 系统能够以单线程的形式运行, 并且运行速度和精度都高于 ORBSLAM2。

参考文献

[1] 尚任. 一种视觉 SLAM 单目半稠密建图方法的实现[J]. 智能计算机与应用, 2021, 11(1): 89–93.

[2] OK K, LIU K, FREY K, et al. Robust object-based slam for high-speed autonomous navigation [C]// 2019 International Conference on Robotics and Automation (ICRA). IEEE, 2019: 669–675.

[3] ZHENG S, WANG J, RIZOS C, et al. Simultaneous Localization and Mapping (SLAM) for Autonomous Driving [C]//IGNSS Conference. 2020: 5–7.

[4] 王霞, 左一凡. 视觉 SLAM 研究进展[J]. 智能系统学报, 2020, 15(5): 825–834.

[5] CHATTERJEE A, GOVINDU V M. Efficient and robust large-scale rotation averaging[C]//Proceedings of the IEEE International

Conference on Computer Vision. 2013: 521–528.

[6] 范涵奇, 吴锦河. 基于 Laplace 分布的双目视觉里程计[J/OL]. 自动化学报: 1–11 [2021–02–22]. <https://doi.org/10.16383/j.aas.c190860>.

[7] KRETZSCHMAR H, STACHNISS C, GRISETTI G. Efficient information-theoretic graph pruning for graph-based SLAM with laser range finders[C]//2011 IEEE/RSJ International Conference on Intelligent Robots and Systems. IEEE, 2011: 865–871.

[8] JOHANNSSON H, KAESS M, FALLON M, et al. Temporally scalable visual SLAM using a reduced pose graph[C]//2013 IEEE International Conference on Robotics and Automation. IEEE, 2013: 54–61.

[9] ZENG T, SI B. A brain-inspired compact cognitive mapping system[J]. Cognitive Neurodynamics, 2020: 1–11.

[10] LATIF Y, NEIRA J. Go straight, turn right: Pose graph reduction through trajectory segmentation using line segments [C]//2013 European Conference on Mobile Robots. IEEE, 2013: 144–149.

[11] KEOGH E, CHU S, HART D, et al. An online algorithm for segmenting time series[C]//Proceedings 2001 IEEE international conference on data mining. IEEE, 2001: 289–296.