软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(6):3052−3068 [doi: 10.13328/j.cnki.jos.006921]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

高清几何缓存多尺度特征融合的渲染超分方法*

张浩南, 过 洁, 覃浩宇, 傅锡豪, 郭延文

(计算机软件新技术国家重点实验室 (南京大学), 江苏 南京 210023)

通信作者: 过洁, E-mail: guojie@nju.edu.cn

摘 要: 人们对图像显示设备高分辨率和逼真视觉感知的需求随着现代信息技术的发展日益增长, 这对计算机软硬件提出了更高要求, 也为渲染技术在性能与工作负载上带来更多挑战. 利用深度神经网络等机器学习技术对渲染图像进行质量改进和性能提升成为了计算机图形学热门的研究方向, 其中通过网络推理将低分辨率图像进行上采样获得更加清晰的高分辨率图像是提升图像生成性能并保证高清细节的一个重要途径. 而渲染引擎在渲染流程中产生的几何缓存 (geometry buffer, G-buffer) 包含较多的语义信息, 能够帮助网络有效地学习场景信息与特征, 从而提升上采样结果的质量. 设计一个基于深度神经网络的低分辨率渲染内容的超分方法. 除了当前帧的颜色图像, 其使用高分辨率的几何缓存来辅助计算并重建超分后的内容细节. 所提方法引入一种新的策略来融合高清缓存与低清图像的特征信息, 在特定的融合模块中对不同种特征信息进行多尺度融合. 实验验证所提出的融合策略和模块的有效性, 并且, 在和其他图像超分辨率方法的对比中, 所提方法体现出明显的优势, 尤其是在高清细节保持方面.

关键词: 神经网络; 渲染; 图像超分; 几何缓存; 特征融合

中图法分类号: TP391

中文引用格式: 张浩南, 过洁, 覃浩宇, 傅锡豪, 郭延文. 高清几何缓存多尺度特征融合的渲染超分方法. 软件学报, 2024, 35(6): 3052–3068. http://www.jos.org.cn/1000-9825/6921.htm

英文引用格式: Zhang HN, Guo J, Qin HY, Fu XH, Guo YW. Super-resolution Method for Rendered Contents by Multi-scale Feature Fusion with High-resolution Geometry Buffers. Ruan Jian Xue Bao/Journal of Software, 2024, 35(6): 3052–3068 (in Chinese). http://www.jos.org.cn/1000-9825/6921.htm

Super-resolution Method for Rendered Contents by Multi-scale Feature Fusion with High-resolution Geometry Buffers

ZHANG Hao-Nan, GUO Jie, QIN Hao-Yu, FU Xi-Hao, GUO Yan-Wen

(State Key Laboratory for Novel Software Technology (Nanjing University), Nanjing 210023, China)

Abstract: With the development of modern information technology, people's demand for high resolution and realistic visual perception of image display devices has increased, which has put forward higher requirements for computer software and hardware and brought many challenges to rendering technology in terms of performance and workload. Using machine learning technologies such as deep neural networks to improve the quality and performance of rendered images has become a popular research method in computer graphics, while upsampling low-resolution images through network inference to obtain clearer high-resolution images is an important way to improve image generation performance and ensure high-resolution details. The geometry buffers (G-buffers) generated by the rendering engine in the rendering process contain much semantic information, which help the network learn scene information and features effectively and then improve the quality of upsampling results. In this study, a super-resolution method for rendered contents in low resolution based on deep neural networks is designed. In addition to the color image of the current frame, the method uses high-resolution G-buffers to assist in the calculation and reconstruct the high-resolution content details. The method also leverages a new strategy to fuse the features of high-resolution buffers and low-resolution images, which implements a multi-scale fusion of different feature information in a specific fusion

* 基金项目: 国家自然科学基金 (61972194, 62032011); 江苏省自然科学基金 (BK20211147)
收稿时间: 2022-08-20; 修改时间: 2022-10-08; 采用时间: 2023-02-15; jos 在线出版时间: 2023-08-09
CNKI 网络首发时间: 2023-08-10

module. Experiments demonstrate the effectiveness of the proposed fusion strategy and module, and the proposed method shows obvious advantages, especially in maintaining high-resolution details, when compared with other image super-resolution methods.

Key words: neural network; rendering; image super-resolution; geometry buffer; feature fusion

1 绪 论

在 3D 游戏、特效影视等场景下人们对高清图像和逼真视觉体验的需求日益增长, 这对渲染技术和显示设备带来了巨大的挑战. 为降低渲染的性能开销, 同时保证高画质, 越来越多的工作提出通过低分辨率图像渲染和上采样到高分辨率的方法来满足性能需求. 随着以深度学习为代表的人工智能技术在各领域的广泛应用, 利用深度神经网络等机器学习技术来完成渲染超采样或图像超分过程成为一种新的趋势[1−3]. 同时工业界也对相关技术进行开发与探索, 如虚幻引擎的 TAAU (temporal anti-aliasing upsample)[4]和 NVIDIA 的 DLSS (deep learning super sampling)[5]. 除此以外, 其他一些用于改善渲染的超采样方法与技术[6−8]致力于平衡渲染时间开销与画面效果两者的关系, 在低成本的条件下尽可能提升图像质量.

目前大部分渲染引擎在工作流程中可以生成携带逐像素属性的几何缓存 (geometry buffer, G-buffer), 这是一般视频流图像中无法获得的额外信息. 即使在高分辨率下, 几何缓存的计算成本仍然较低 (通常一帧仅几毫秒). 这种特性使其相比需要繁重工作负载的着色、后处理等环节有了独特的性能优势. 近年来许多研究方法[9−11]将含有逐像素属性的几何缓存等额外信息用于渲染去噪, 图像重建等工作. 受这些工作启发, 我们采用高分辨率几何缓存来指导低分辨率图像的超分工作, 能有效提高结果图像的逼真度. 即便如此, 如果将逐像素属性的几何缓存以连接或相加的简单方式作为辅助, 这样做容易让预测过程对不同信息产生混淆并使得神经网络在维度和空间区域上学习到错误的知识, 例如过度依赖几何缓存中的特征, 并在相应区域生成错误的像素值, 最终形成伪影等视觉错误. 因此, 针对两者分别提取出的特征处理方式会成为影响结果表现的重要因素. 合适的特征融合方式可以帮助神经网络更好地处理图像特征与几何缓存特征的关系, 使得模型能更好地区分不同特征信息的重要性并加以利用. 针对图像超分过程, 我们使用几何缓存时需要剔除无效信息并选择有效信息, 从而填补低清图像中的缺失内容并有效提高网络预测结果的质量.

综上分析, 为平衡渲染过程中的资源消耗与图像画质, 我们在几何缓存额外信息的辅助下为渲染内容设计一种图像超分技术, 使得低分辨率图像可以通过神经网络在给定超分比例下预测得到高保真度的高清图像, 从而降低渲染高清图像的时间开销. 加入我们方法的具体渲染管线见图 1, 我们先在输出的高分辨率下生成深度、法向等几何缓存, 再设置为低分辨率并进行常规渲染, 利用生成的低清图像和高清几何缓存进行网络推理, 得到的高清图像可以进行抗锯齿等后处理操作. 为了保证预测图像的质量, 我们给出了一种针对高清缓存与低清图像的特征信息的融合策略, 该策略利用我们设计的融合模块对两种特征信息进行不同尺度上的融合, 对高分辨率几何缓存中提取的特征进行调整与辨别, 在保证光照, 阴影等画面效果尽可能不被几何缓存信息错误重建的条件下, 选取其中对超分过程有辅助作用的信息对低清图像提取的特征进行合理的填充修复, 从而提升图像超分的质量. 在一些需要大量计算 (例如, 开启实时光线追踪) 场景下, 相对于传统渲染流程, 使用我们的方法辅助渲染可以省去昂贵的高清帧渲染代价. 同时, 相比于如视频内插等其他需要相邻帧或运动信息的超分方法[12−15], 我们的方法仅依赖当前帧的相关信息, 这样做有利于处理如光源闪烁变化等画面剧烈变化的情况, 从而提升画面预测的鲁棒性.

综上所述, 我们的贡献如下.

(1) 设计了一种针对渲染内容的超分神经网络, 利用低分辨率图像和高分辨率的几何缓存来执行渲染图像超分任务, 能够在保证高清画质的前提下显著提升渲染高分辨图像的性能, 满足实时图形应用的需求.

(2) 提出一种针对高清几何缓存和低清渲染图像的多尺度特征融合方案, 使得几何缓存特征与内容图像特征在不同尺度上实现多次融合, 从而有效利用高清几何缓存中的关键信息补充低清渲染图像中确实的内容细节.

(3) 相对于之前的超分工作, 我们的方法不需要历史帧和相关运动信息, 从而可以处理任意剧烈变化的场景 (如场景中光源的闪烁变化) 且无论在视觉感知和图像质量的定量分析方面均显著提升.

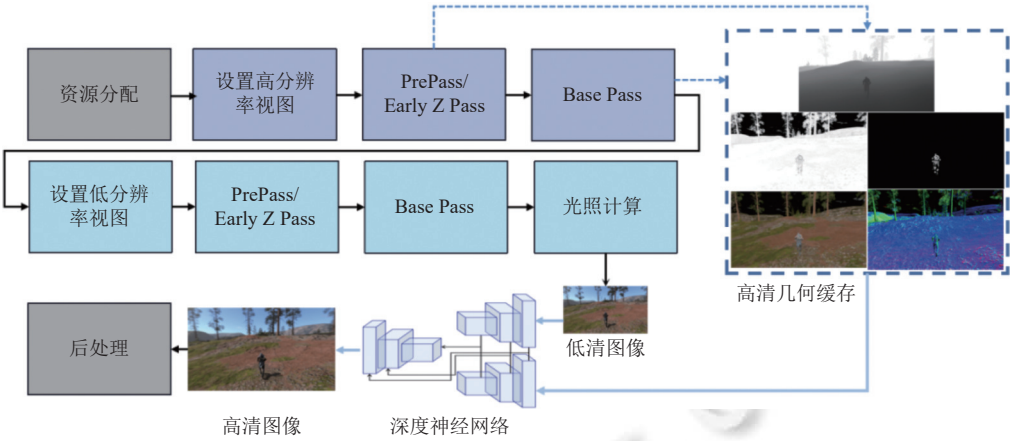


图 1 融合本文提出的超分技术的实时渲染流程 (基于虚幻引擎)

2 相关工作

2.1 基于单张图像的图像超分辨率

传统的图像超分辨率方法通常采用最近邻插值、双线性插值、双三次插值等经典算法. 这些算法虽然简单且速度快, 但对图像视觉效果的改善程度有限, 通常很难恢复图像细节. 近些年来, 与基于传统方法相比, 基于深度学习的解决方案在峰值信噪比 PSNR 等指标衡量与视觉感知方面都表现出更加优异的性能. 如 2014 年由 Dong 等人进行超分重构时, 提出了低清图像插值放大和 CNN 补充内容细节的方法[16], 文中提出的 SRCNN 网络是深度学习用在超分重建上的最早期的工作. EDSR 网络[17]是 NTIRE2017 超分辨率挑战赛的冠军方案, 该方案移除了残差结构中一些不必要的结构, 对网络进行了精简, 同时提出了一种多尺度深度超分系统, 其在处理不同超分尺度的情况下均有着较为优越的性能. RCAN 网络[18]则引入了注意力机制来区别对待不同通道的特征, 并提出了一种残差中的残差 (residual in residual, RIR) 结构, 构建了一个很深的神经网络, 具有超过 400 个卷积层, 同时超分预测结果出色. Ledig 等人[19]将 GAN 引入单个图像超分领域, 设计了 SRGAN 网络并使用新的损失函数进行训练, 以提升预测结果的真实感. 丁玲等人在 SRGAN[19]以及常用的 VGG[20]重构损失框架上, 设计了 VGG 能量损失, 并构建 U-Net[21]自编码结构 VGG-UAE 来缓解高分辨率图像输入下输出判别信号不稳定的问题[22]. 潘宗序等人将全局字典学习中利用图像库获取附加信息的思想融入自适应字典学习的过程中, 提出了一种基于自适应多字典学习的单个图像的超分算法[23]. 这些方法使用单张图像作为输入, 而单个低清图像本身包含信息有限, 从而无法恢复原图像向低分辨率的退化过程中丢失的高频信息.

2.2 基于多张图像的图像超分辨率

相比于基于单个图像的超分方法, 多张图像的超分方法多应用于视频等领域, 其引入了时域信息, 并更多利用相邻帧的时空相关性来提升结果性能. 一种常见的研究方向是采用运动补偿和运动估计技术. 其中运动估计可以提取不同帧之间的运动信息, 而运动补偿则利用不同帧之间的运动信息进行图像扭曲, 从而使帧对齐. 例如 Caballero 等人提出的 VESPCN[24]将动作补偿和视频超分联合用以提升性能. NTIRE19 挑战赛冠军网络 EDVR[25]提出了时空注意力融合模块与级联可变形的对齐模块. 相比于需要帧对齐的研究方法, 非对齐方法使网络模型自己学习帧间的相关信息, 利用空间或时空信息进行特征提取与处理. 例如 DUF 网络[2]就避免了显式的运动补偿, 而是使用动态上采样滤波器对输入图像直接进行重构, 并通过残差部分来增加细节内容. Isobe 等人[12]提出一种新的循环残差网络 (recurrent residual network, RRN) 来实现高效的视频超分. S-Lab 与南洋理工大学的研究人员给出的 BasicVSR++[15]引入了二阶网格传播和光流引导可变形对齐, 成为了目前最先进的视频超分网络. 由 Yi 等人[26]提出 PFNL 网络则使用非局部残差块来提取时空特征信息, 取得了比使用运动补偿更好的实验结果. 相比此前的

单张图像超分方法, 基于多张图像的超分辨率方法表现出更强的性能潜力, 但也在时序信息的提取与处理上花费更多, 使得网络结构更复杂, 增加了推理时间, 不适用于实时渲染场景下的低延迟要求.

2.3 基于几何缓存的图像重建

几何缓存是由渲染管道中的中间渲染过程生成的几何体和材质信息的屏幕空间表示形式, 在传统的渲染流程中发挥着重要作用. 广义层面的几何缓存包含法线, 世界空间坐标, 深度等多种属性信息. Lu 等人提出的 DMCR-GAN 是一个具有残差注意力模块与分层特征调制的对抗网络, 通过调整对不同输入辅助缓存 (auxiliary buffer) 的注意力等实现了蒙特卡洛渲染的去噪工作[27]. Nalbach 等人[11]收集渲染中产生的材质, 几何和光照信息来预测屏幕空间中的照明效果. Chakravarty 等人[28]提出的用于重建使用蒙特卡罗方法渲染的图像序列的神经网络同样使用了如深度, 法向等辅助缓存作为输入, 该网络在样本数量少的情况下仍有着合理的运行速度和结果表现. 2021 年 Guo 等人[29]提出的 ExtraNet 使用了反照率, 深度, 法向, 粗糙度, 金属度等几何缓存和历史帧来进行外推帧的预测, 可以在没有明显伪影的条件下产生合理的外推帧结果, 并在实践中成功提升了帧率. 由 Xiao 等人[9]提出的超分神经网络 NSRR 以低清下的当前帧与历史帧的 RGB 图像, 深度图与运动矢量信息为输入, 通过特征提取, 后向扭曲 (backward warping) 等步骤重建高分辨率的图像, 并获得了高保真度和时间稳定的结果. 杜文俊等人[30]提出通过存储的几何缓存和像素分类, 用基于几何辅助分析的反走样算法来精确重构出需要反走样的几何边界. 邵鹏等人[31]提出的 IAAFXAA 算法可以从几何缓存中提取深度和法线信息并借助该信息进行更加精确的检测, 以提高抗锯齿性能, 避免图像过度模糊. 以上方法在内容去噪、图像超分等领域用几何缓存进行辅助, 为本文的研究方法提供了相关思路与借鉴.

3 基于高清几何缓存的渲染内容超分辨率

3.1 问题描述

在不开启多采样抗锯齿的情况下, 渲染器通过对 3D 场景下的所在位置进行点采样并计算形成每个像素. 因此对渲染内容实现高质量的上采样面临的困难是, 低分辨率的输入图像在部分阴影, 几何边缘处等区域发生走样, 对应区域需要还原的高频信息完全丢失. 为了解决这一问题, 许多研究工作[4,5,9]考虑使用时序上连续帧的冗余性来提供更详细的信息来辅助内容重建, 例如 Xiao 等人的方法[9]使用了当前帧和历史帧的像素颜色, 深度图以及运动矢量. 而在渲染帧的内插研究中, 也有使用光流来处理时序信息的工作[32]. 运动矢量虽然精确, 但是无法处理因光照导致的光影变化 (如阴影和高光反射); 光流虽然理论上能处理各种运动效果, 但是精度较差, 而且需要额外的运算代价. 此外, 无论是基于运动矢量的方法还是基于光流的方法, 都假设相邻连续帧之间变化幅度较小. 一旦相邻帧之间出现剧烈变化 (如光源的剧烈闪烁变化), 则时序方法很容易产生问题.

几何缓存的概念来自延迟渲染 (deferred rendering), 其基本思想是先将三维场景中所有物体的几何信息写入屏幕空间的缓冲区中, 等到计算光照并对屏幕空间下各像素着色时, 再将这些信息取出来用于计算. 几何缓存则是指存储这些信息的容器, 一般包含像素对应的位置、法线、漫反射颜色、高光颜色等数据. 几何缓存的生成本质上是对物体执行无光照材质的渲染, 并将各物体的相关信息写入对应的渲染纹理中, 该过程由几何通道 (geometry pass) 负责, 而在后续的光照通道 (lighting pass) 中, 几何缓存中的物体的几何信息会被来执行光照计算, 从而对每个像素进行着色. 几何缓存使得每个光源只需要处理固定大小的缓冲区, 而不受场景中物体数量的限制, 有效降低了计算开销. 由于不涉及复杂的光照、阴影等相关计算, 因此几何缓存的生成成本相比直接渲染画面而言要低很多.

经过以上分析, 我们考虑在单个帧内容图像的基础上, 尽可能使用额外的几何缓存来帮助特征提取和图像重建. 在本次研究工作中, 除了低分辨率的内容图像, 我们选择了共 9 个通道的 5 种几何缓存 (如图 2 所示) 来帮助超分过程中的内容重建, 包括:

(1) 底色 (base color), 共有 3 个通道, 定义了材质的整体颜色. 底色是用来记录颜色信息的材质贴图, 它记录剔除光影变化后的基础颜色, 将其作为输入能帮助网络确定预测图像中材质表面所在像素的基准色调.

(2) 场景深度 (scene depth), 共有 1 个通道, 存储了场景中各像素的深度值. 场景深度有助于辨别物体轮廓与屏幕空间中像素对应的前后关系, 从而对更接近镜头的物体着重预测更多的细节.

(3) 粗糙度 (roughness), 共有 1 个通道, 描述各像素的表面粗糙程度, 其决定了反射的模糊或清晰度.

(4) 金属色 (metallic), 共有 1 个通道, 描述各像素的表面在多大程度上“像金属”, 非金属的对应值为 0, 金属的对应值为 1. 粗糙度与金属色则对高光区域有影响, 例如在粗糙度较大的区域所预测出的高光应该是大且暗淡的, 从而帮助网络不会被低清图像中的走样而对高光进行错误估计.

(5) 法向 (world normal), 共有 3 个通道, 存储了场景中各像素的法线值, 其影响着光线反射, 折射后的着色结果. 法向能辅助模型识别凹凸不平的表面, 也即法向变化强烈的区域, 并在超分预测时展现合理的表面细节.

(a) 底色 (b) 场景深度 (c) 粗糙度 (d) 金属色 (e) 法向

图 2 本文方法所采用的几何缓存

由于几何缓存并非真实的渲染图像, 过于简单的处理方式会导致最终预测结果的部分区域出现颜色偏差以及不合理的高光、阴影. 因此, 如何有效利用这些包含物体轮廓、表面纹理等丰富高频信息的几何缓存是方法设计中需要考虑的问题. 为此, 我们提出了一种针对高清几何缓存特征和低清图像特征的新的融合方法, 从而提升超分结果的保真度. 基于这个融合方法, 我们设计了一个端到端的高性能神经网络, 能够进行实时的渲染图像超分.

3.2 方法设计

如图 3 所示, 我们网络整体划分为 3 个子网络: 特征提取, 特征融合与解码部分. 各网络层下方数字为该层的输出通道数, 图 3 中粉色标注卷积层的卷积核大小为 3×3. 转置卷积层和绿色标注卷积层的卷积核大小为 2×2, 步长为 2. 不同部分在图 3 中带颜色的虚线框进行划分. 我们用 I_{LR}, I_G, I_{HR} 分别表示作为输入的低清内容图像, 几何缓存信息与最终输出的高清内容图像.

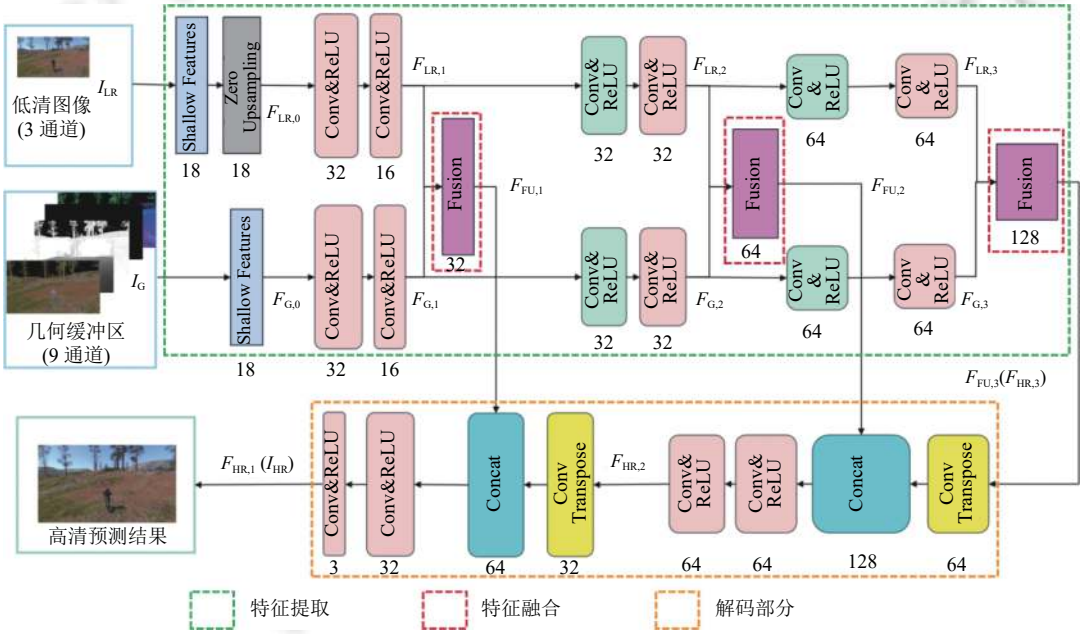


图 3 整体网络架构

3.2.1 特征提取

为了捕获内容图像与几何缓存中的细节与空间信息, 我们需要对输入进行低层次的特征提取, 并完成对低清图像特征的上采样, 以便让两种特征在同一分辨率空间下完成后续操作. 首先我们分别从 I_{LR}和I_{G} 中提取浅层特征 $F_{\mathrm{LR},0}$和$F_{\mathrm{G},0}$:

$$\begin{cases} F_{\mathrm{LR},0}=H_{\mathrm{UP}}(H_{\mathrm{SF}}^{\mathrm{LR}}(I_{\mathrm{LR}})) \\ F_{\mathrm{G},0}=H_{\mathrm{SF}}^{G}(I_{\mathrm{G}}) \end{cases},$$

其中, H_{UP} 为零上采样操作[9], 用于将低分辨率内容特征上采样至所需的高分辨率空间. 零上采样将图像中每个像素分配到高分辨率下对应的 s 个像素中的固定一处, 而剩下 $s-1$ 个像素会被置为 0, 其中 s 为上采样比值. 用 h,w 分别表示低分辨率图像的高和宽, $I\in R^{h\times w}$, $O=H_{\mathrm{UP}}(I)\in R^{(sh)\times(sw)}$ 分别表示该方法的输入图像与输出图像, 则两者内像素映射关系为:

$$O(y,x)=\begin{cases} I(y/s,x/s), & s|x\wedge s|y \\ 0, & \text{otherwise} \end{cases},$$

其中, $(y,x)\in\{0,1,\ldots,sh-2,sh-1\}\times\{0,1,\ldots,sw-2,sw-1\}$. 注意以上公式的输入输出均为单通道图像, 面向多通道特征时同样适用, 只需对各通道逐一操作. 零上采样操作考虑将低清空间下各像素映射到高清空间下某一位置, 其余位置可以视为“未被着色”的空像素, 这提升网络针对这些有效或无效信息的辨别能力, 并在后续融合操作中利用几何缓存特征与有效原像素对这些无效信息进行填充着色, 从而提升了模型预测图像的保真度.

$H_{\mathrm{SF}}^{\mathrm{LR}},H_{\mathrm{SF}}^{\mathrm{G}}$ 各自包含 3 个卷积-ReLU 激活操作与连接操作. 浅层特征提取模块的更多细节如图 4 所示, 该部分中卷积核大小为 3×3, $C_{\mathrm{in}},C_{\mathrm{out}}$ 的取值根据情况讨论. 针对 $I_{\mathrm{LR}},I_{\mathrm{G}}$ 的两个模块结构基本一致, 但在最后一次卷积的输出通道数有差别. 为满足后续的融合操作, 从内容图像与几何缓存提取出的浅层特征通道数必须相等, 而前者输入为 3 通道, 后者输入为 9 通道, 所以在进行最后一次卷积时, 针对内容图像的卷积输出通道为 15, 而针对几何缓存的卷积输出通道为 9. 经过 3 层卷积-ReLU 激活的结果与输入本身连接, 并得到 18 通道的浅层特征.

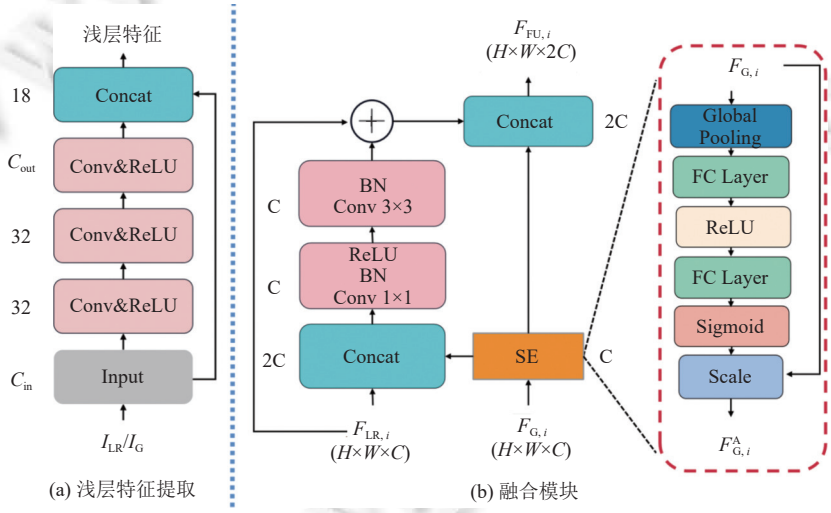


图 4 浅层特征 (shallow features) 提取模块结构与融合模块

在后续步骤中我们参考了 U-Net[21]的设计思想, 采用不同层次的编码-解码操作来对高清图像进行重建. 第 i 层次的编码器部分对应输入为 $F_{\mathrm{LR},i-1}/F_{\mathrm{G},i-1}$, 通过该层次编码器得到输出 $F_{\mathrm{LR},i}/F_{\mathrm{G},i}$:

$$\begin{cases} F_{\mathrm{LR},i}=H_{EN,i}^{\mathrm{LR}}(F_{\mathrm{LR},i-1}) \\ F_{\mathrm{G},i}=H_{EN,i}^{\mathrm{G}}(F_{\mathrm{G},i-1}) \end{cases},$$

其中, $H_{EN,i}^{\mathrm{LR}},H_{EN,i}^{\mathrm{G}}$ 分别表示第 i 层次中针对内容图像特征和几何缓存的编码器, 每个编码器内包含 2 个卷积-ReLU

操作, 为适应渲染场景的需要, 我们选择 3 个大小层次的编码, 即 i 的取值为 1, 2 或 3. 卷积运算时我们通过步长和卷积核大小控制中间特征的尺寸, 同时更小尺寸的中间特征需要更多的通道数来保证提取足够多的信息, 更多细节在图 3 中展示.

3.2.2 特征融合

内容图像与几何缓存之间具有相关性, 高分辨率的几何缓存中得出的丰富特征和物体轮廓边界信息对低清图像的超分具有辅助作用. 这种作用是有向的, 即从几何缓存提取到的特征信息单向辅助内容图像的超分过程. 同时多层次化融合的引入能在不同尺度上使内容图像与几何缓存的特征能够更有效地融合. 我们的具体操作为: 将第 i 层次中编码器计算得到的中间特征 $F_{\mathrm{LR},i}, F_{\mathrm{G},i}$ 作为输入, 经过该层次的融合操作 $H_{\mathrm{FU},i}$ 后得到输出 $F_{\mathrm{FU},i}$:

$$F_{\mathrm{FU},i} = H_{\mathrm{FU},i}(F_{\mathrm{LR},i}, F_{\mathrm{G},i}) = H_{\mathrm{Res},i}(F_{\mathrm{LR},i}, F^{A}_{\mathrm{G},i}).$$

特征融合部分更多细节见图 4, 图中 FC Layer 表示全连接层, BN 表示 batch normalization, ⊕ 表示逐元素相加, SE 模块中的参数 reduction 被设置为 16, 各层旁数字表示该层通道数. 在特征融合模块中, $F^{A}_{\mathrm{G},i}$ 会进行 SE (squeeze-and-excitation) 操作 [33], 这是因为几何缓存特征的不同通道包含不同的信息, 我们需要选择有助于补充本层次内容图像细节的信息, 也就是说, 网络需要引入注意力机制, 从而更多关注 $F^{A}_{\mathrm{G},i}$ 中的某些通道. $F^{A}_{\mathrm{G},i}$ 会先经过压缩 (squeeze) 操作, 利用全局均值池化将各个通道下的特征编码为全局特征量, 然后将全局特征量通过以 Sigmoid 函数为激活函数的两个全连接层, 完成激活 (excitation) 并得到激活值, 最终将激活值与 $F_{\mathrm{G},i}$ 相乘得到 $F^{A}_{\mathrm{G},i}$. 整个操作调整了几何缓存不同特征的权重, 从而使得后续操作中更容易辨别能够辅助超分的特征信息.

SE 操作的结果 $F^{A}_{\mathrm{G},i}$ 与 $F_{\mathrm{LR},i}$ 进行连接, 然后经过一个受 ResNet[34]启发的残差块 $H_{\mathrm{Res},i}$, 该残差块包含一个 1×1 的卷积操作和一个 3×3 的卷积操作, 其生成的残差结果在与 $F_{\mathrm{LR},i}$ 相加后可以对内容图像特征进行补充. 一种直观解释是将 $F_{\mathrm{LR},i}, F^{A}_{\mathrm{G},i}$ 视为 $H\times W\times C$ 空间下的函数表示, 其中 $F_{\mathrm{LR},i}$ 是上采样后得到的低清内容图像特征, 包含丰富的低频信息但缺少高频信息. 我们希望计算得到具有丰富高频特征的高清内容图像特征表示, 而 $F^{A}_{\mathrm{G},i}$ 本身从高分辨率空间的几何缓存提取得到, 在边缘, 角点等高频特征上更为准确, 所以其有助于定位和修正 $F_{\mathrm{LR},i}$ 中的无效信息. 利用残差块有利于信息在不同层之间相互传导的特性, 我们将连接输入用 1×1 的卷积进行降维处理, 将 $2C$ 通道数降为 C 通道数并以 ReLU 函数为激活, 再用一层卷积实现对 $F_{\mathrm{LR},i}$ 函数表示的修正过程, 使其更接近高清内容图像特征表示. 残差块处理后的信息再与 $F^{A}_{\mathrm{G},i}$ 连接, 即为融合操作的最终结果 $F_{\mathrm{FU},i}$.

进一步来说, 我们可以把几何缓存与渲染图像看成像素分布, 而这两种分布显然是不相同的, 例如在后续光照计算中得到的高光、阴影以及材质颜色变化并不会在基色、深度的像素值中得到体现, 并且低清图像采样率不足的问题也会导致从中提取到的两种特征会存在分布上的差异. 我们希望借助几何缓存分布的信息, 用低清图像最终重建出高清渲染图像, 而如果使用相加操作, 则从几何缓存与渲染图像提取到的特征在融合时会引起分布上的直接混合, 却无视了两者在均值、方差等方面存在的差异, 从而使得几何缓存特征在超分过程与不同分布的图像特征产生混淆, 偏离了“重建高清图像像素分布”的目标. 连接操作的问题类似, 该操作将两者直接视为同一特征的不同通道并进行后续计算, 几何缓存特征代表的分布无法通过注意力调制等手段达成合适的形态, 从而使特征在解码时错误选择不同通道的信息. 相比连接和相加操作, 我们使用的融合方法首先进行了 SE 操作, 实现了 $F_{\mathrm{G},i}$ 不同通道间的注意力分配, 这可以看作是对几何缓存特征的初步调节, 而后续的残差块提升了融合模块的特征提取能力, 对两个不同特征进一步调节与提取信息. 因此 SE 操作让最终与 $F^{A}_{\mathrm{G},i}$ 连接得到的特征在不同通道间能达成更为合理的像素分布, 并且含有连接-卷积的残差块比直接相加有更强的特征表达与调节能力.

本文提出的网络结构中共有 3 个针对不同尺度的特征融合模块, 对应的融合结果分别为 $F_{\mathrm{FU},1}, F_{\mathrm{FU},2}, F_{\mathrm{FU},3}$. 多尺度的融合机制可以让网络针对不同形态的信息特征进行融合, 充分利用高清几何缓存中的高频细节, 从而实现更为真实的画面效果. 这些融合结果会传递到解码部分, 以连接的方式帮助解码器更好地对图像进行超分重建.

3.2.3 解码部分

对应此前 3 个层次的编码部分, 解码部分包含 2 个跳跃连接 (skip-connection). 从第 3 层次的融合特征开始解

码, 每层次解码过程将后一层次的结果作为输入, 先经过一次转置卷积操作, 用以扩张特征图尺寸和降低通道数, 再通过跳跃连接与对应层次的融合特征进行连接, 最后经过两层卷积得到该层次解码的输出结果:

$$F_{\mathrm{HR},i}=H_{\mathrm{DE},i}(F_{\mathrm{HR},i+1},F_{\mathrm{FU},i})=H_{\mathrm{DE\text{-}Convf},i}([ConvTrans_{\mathrm{DE},i}(F_{\mathrm{HR},i+1}),F_{\mathrm{FU},i}]),$$

其中, $F_{\mathrm{HR},i+1},F_{\mathrm{HR},i}$ 分别表示第 i 层次解码器的输入 (第 $i+1$ 层次的输出) 和输出 ($i=1,2,3$), 特别地, $F_{\mathrm{HR},3}=F_{\mathrm{FU},3}$. 另外 $H_{\mathrm{DE},i}$ 表示第 i 层次解码器运算, $H_{\mathrm{DE\text{-}Convf},i}$ 将连接结果进行 2 次 3×3 大小的卷积操作, 并得到输出结果. $ConvTrans_{\mathrm{DE},i}(\cdot)$ 表示第 i 层次解码器中的转置卷积操作. $[\cdot,\cdot]$ 表示对两个特征的连接操作. 关于卷积输出通道数等细节见图 3. 最终预测的高清内容图像为第 1 层次的解码结果, 即 $I_{\mathrm{HR}}=F_{\mathrm{HR},1}$.

总的来说, 我们设计的方法利用 3 个尺寸规模上的“编码-解码”操作提升渲染内容图像的超分质量, 但在扩张路径部分, 我们让内容图像和几何缓存的特征进行相似而权值独立的卷积操作, 以此帮助模型区分和学习两者在图像超分过程中的不同作用. 浅层特征的提取对内容图像与几何缓存进行低层次的特征提取, 增加了网络深度, 从而使得其表达能力增强. 我们设计的融合模块能够帮助网络定位几何缓存特征中对超分有辅助作用的信息与图像特征中需要补充的缺失内容, 并且利用不同尺度进行多层次融合, 使得超分重建过程能更好地恢复图像细节.

3.3 损失函数

在网络训练环节中我们选择了结构相似性指数度量 (structural similarity index measure, SSIM)[35]与感知损失的线性组合作为损失函数, 其中感知损失函数是学习感知图像块相似度 (learned perceptual image patch similarity, LPIPS)[36], 其利用预训练好的网络模型提取特征来度量两张图像之间的感知差异, 度量值越低表示两张图像越相似, 反之则差异越大. 对于该感知损失函数, 我们选择的预训练网络模型为 VGG-16[20]. 网络训练时我们的损失函数为:

$$L(P,T)=1-D_{\mathrm{SSIM}}(P,T)+w\cdot D_{\mathrm{LPIPS}}(P,T),$$

其中, P, T 分别表示高分辨率的预测内容图像与真值图像, 权值系数 $w=0.2$. D_{SSIM} 表示两图像之间的 SSIM 度量值. D_{LPIPS} 表示 LPIPS 度量值, 将两图像分别输入预训练的 VGG-16[20]网络模型, 获取网络各层输出并计算对应通道的余弦距离, 最后对各距离进行平均, 得到度量值.

4 实验分析

4.1 实验数据与训练细节

为了训练和测试模型, 我们将虚幻引擎 4 作为渲染引擎并引入虚幻引擎市场 (https://unrealengine.com/marketplace) 中的 3 个场景构建了一个大规模数据集. 其中这些场景涵盖广泛的着色效果, 如光线反射, 软阴影, 不同光源与复杂遮挡等等, 各场景的示例帧见图 5. 我们的数据集由每个场景下 2 500 帧的训练集和 500 帧的测试集组成. 数据集中的内容图像未经过任何后处理 (包括抗锯齿), 同时我们选择在各个场景关闭运动模糊 (motion blur), 这样做可以降低镜头随人物与场景运动时产生的背景模糊.

(a) Bunker (b) Redwood Forest (c) Medieval Docks

图 5 作为数据集的 3 个场景的示例帧

我们选择了 4×4 规模的超分来训练和测试网络, 数据集中真值图像和几何缓存的分辨率为 1920×1080, 输入图像的分辨率为 480×270. 网络在训练时会将数据集中的输入图像, 真值图像和几何缓存进行 3×3 分块. 如前所述, 本次实验共有 3 个场景数据集, 针对每个场景的数据集各训练 1 个模型. 所有训练和测试均在配有一块 NVIDIA RTX 3090 GPU 的电脑上进行.

我们的网络使用 PyTorch 框架[37]实现. 训练优化过程中使用小批量随机梯度下降法与 Adam 优化器[38]. 在训练参数方面, 我们设置批量大小 (batch size) 为 8, Adam 优化器的 $\beta_1 = 0.9, \beta_2 = 0.999$, 初始学习率为 1E−3, 且每经过一个 epoch 学习率衰减至 0.98 倍, 共训练 100 个 epoch. 训练时网络的初始化使用了 Xavier 初始化方法[39].

4.2 与现有方法的比较

在本节, 我们的方法与几个有代表性的超分辨率研究工作进行了对比, 其中包括单个图像超分方法 RCAN[18], 渲染图像超分方法 NSRR[9], 以及视频超分方法 RRN[12]. 我们将对比方法进行了代码复现, 并在相同的数据集上进行训练和测试. 我们调整了 RRN[12]的数据集输入, 使网络输入的低清图像是我们在低分辨率下渲染得到的场景图像, 而不是将高清图像用高斯核模糊与下采样操作获得输入图像.

我们用两个指标来评估结果: 峰值信噪比 PSNR 与结构相似性指数度量 SSIM[35], 两者均是在单个图像评估工作中重要的度量指标, 且数值越高代表结果图像质量越好.

我们计算了各方法中 PSNR 和 SSIM 指标, 统计方式为各个场景测试集下所有测试图片的平均值. 表 1 对上述指标进行了比较. 我们可以从中看出, 我们的方法在不同数据集上的两种度量指标均优于其他方法. 除指标外, 表 1 还给出了我们方法与对比方法单帧超分的推理时间对比, 其中超分规模均为 4×4, 目标分辨率为 1920×1080. 从中可以看出, 我们方法的性能对比两个计算机视觉方法[12,18]有明显优势, 与 NSRR[9]运行时间相近, 但我们的预测结果有着更逼真的画面表现.

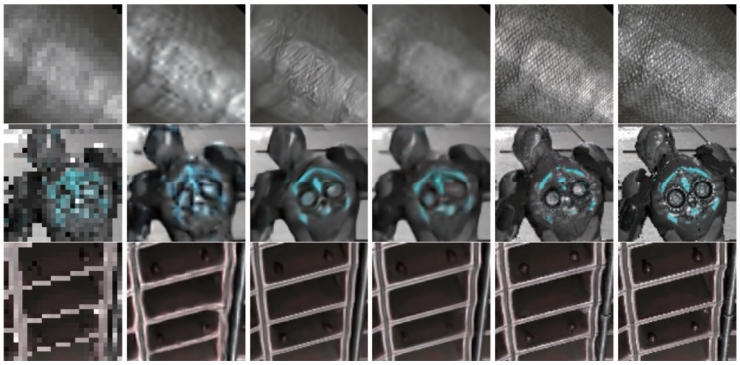
表 1 我们的方法与其他方法的性能指标与运行时间对比

指标	场景	RCAN	NSRR	RRN	Ours
PSNR (dB)	Bunker	26.54	24.88	27.79	28.42
	Redwood Forest	18.68	17.25	19.98	24.25
	Medieval Docks	21.24	20.79	22.82	25.37
SSIM	Bunker	0.796 6	0.760 4	0.811 4	0.904 1
	Redwood Forest	0.391 0	0.422 8	0.407 1	0.871 2
	Medieval Docks	0.684 0	0.700 6	0.705 8	0.870 5
运行时间 (ms)	所有场景	41.52	24.42	93.17	23.04

图 6–图 8 给出了 3 个场景下各方法的预测结果, 其中 Input 表示输入的低清图像, Ours 表示我们方法的超分结果, GT 表示真值图像. 从图中可以看到, 我们方法的超分预测结果在视觉效果上比其他方法更为逼真, 尤其在树丛, 细密金属外皮等具有丰富纹理的区域, 低分辨率下的当前帧和历史帧的相关信息的还原效果较差, 这是因为即使这些细微部分出现在了时序上连续的若干帧中, 其仍然无法提供足够的高频信息来为超分提升结果质量. 而我们的方法使用高分辨率的几何缓存而不是历史帧信息, 能够帮助网络对这些区域的物体轮廓和表面纹理提供参考从而进行合理预测, 同时网络使用我们的融合模块并对特征进行多尺度融合, 对不同信息进行有效选择与剔除, 从而一定程度上避免了过度依赖几何缓存等引起的视觉错误, 提高了针对预测画面的感知真实. 因此我们也能够解释表 1 中在 Redwood Forest 场景下我们方法的指标度量显著高于其他方法, 这是因为该场景下树丛, 草丛较多, 具有复杂的几何轮廓和背景.

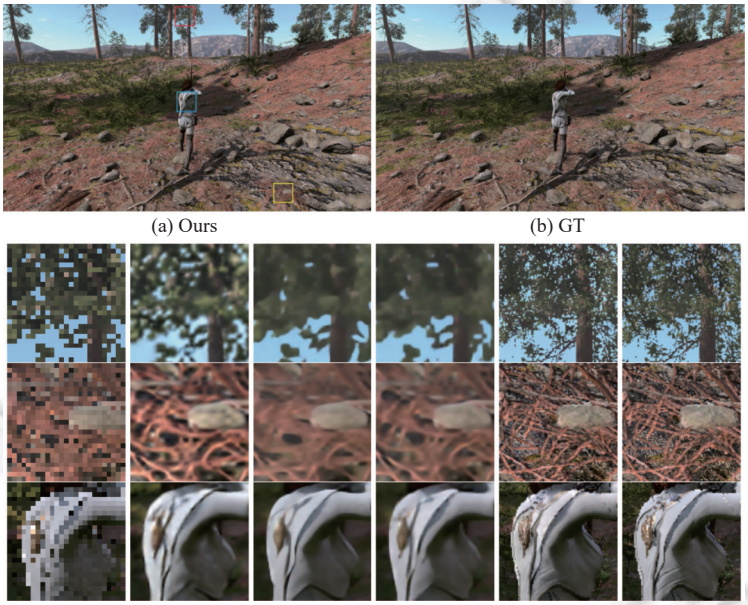
(a) Ours (b) GT

图 6 Bunker 场景下的我们方法与其他方法的视觉结果对比



(c) Input (d) NSRR (e) RCAN (f) RRN (g) Ours (h) GT

图 6 Bunker 场景下的我们方法与其他方法的视觉结果对比 (续)



(a) Ours (b) GT

(c) Input (d) NSRR (e) RCAN (f) RRN (g) Ours (h) GT

图 7 Redwood Forest 场景下的我们方法与其他方法的视觉结果对比

进一步来说, 以图 8 中第 2 行各方法对于远处船舶上的缆绳为例, 低分辨率的输入图片在该区域表现为零散的黑色像素, 混叠现象严重, 原有的绳索结构几乎完全损坏. RCAN[18]和 RRN[12]方法均无法对此进行有效重建. NSRR[9]网络对该区域进行了一定程度的还原, 但在连续性有所欠缺, 最终呈现出“杂乱”的画面. 而我们的方法能利用几何缓存与合理的融合框架对细节进行有效恢复, 产生与参考图像极其相近的结果.

(a) Ours (b) GT

图 8 Medieval Docks 场景下的我们方法与其他方法的视觉结果对比

图 8 Medieval Docks 场景下的我们方法与其他方法的视觉结果对比 (续)

特别地, 图 9 给出了一个 Bunker 场景下的特殊示例, 即背景光颜色闪烁变化, 当前帧及其相邻帧的光源颜色由白、黄、蓝、红随机变化. 如图 9 右侧所示, 第 2 行的 NSRR[9]结果在对应区域的水管表面表现为偏黄色的底色, 第 3 行的 RRN[12]结果在对应区域呈现不规则的蓝色伪影. 这两处错误在当前帧图像中均无对应画面效果, 而是相关方法从时序信息中获取并使用了对应的颜色信息. 相比较而言, 我们的方法仅使用当前帧的画面内容与几何缓存, 不会从相邻帧中学习错误的内容, 从而能够处理光源变化剧烈等极端场景.

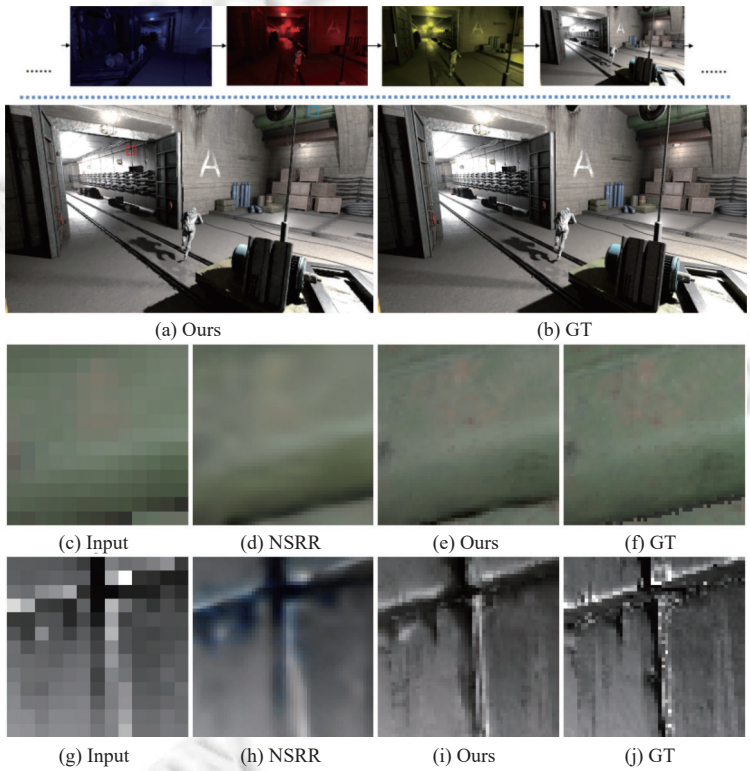


图 9 光源闪烁变化场景示例 (第 1 行) 以及我们方法与其他方法的视觉结果对比 (后 3 行)

4.3 运行时间分析

表 2 提供了不同场景生成几何缓存和内容图像以及执行我们网络推理的运行时间, 表中数据以 ms 为单位.

几何缓存在不同场景下生成的时间花销通常是不同的, 其与运行时场景几何图形的复杂性密切相关. 由表 2 可以看出, 生成几何缓存的时间花销远低于同分辨率下渲染内容图像的时间开销, 在我们使用的 3 个场景中, 几何背景较为复杂的 Redwood Forest 场景生成所需的几何缓存平均仅为 1.40 ms/帧, 而直接渲染生成高分辨率图像花销较大. 因此在一定条件下, 生成低清图像与高清几何缓存, 并用网络进行推理的开销会比直接渲染高清图像更为低廉. 这也是本次研究的重要动机.

表 2 不同场景生成几何缓存和内容图像的运行时间 (ms)

数据/场景	Bunker	Redwood Forest	Medieval Docks
几何缓存 (1920×1080)	0.11	1.40	0.20
低清图像 (480×270)	5.18	4.37	9.75
网络推理 (4×4)	23.04	23.04	23.04
总计	28.33	28.81	32.99
高清图像 (1920×1080)	40.58	14.21	98.62

注: 所有数据在NVIDIA RTX 3090上测量

我们将网络转为 ONNX 模型[40]并使用 16 位精度的 NVIDIA TensorRT[41]进行加速. 从表 2 中可以看出, 我们的方法基本独立于场景复杂度, 对不同场景下产生高清帧的平均时间开销稳定在 30 ms 附近, 能达到大约 30 FPS 的帧率, 满足实时渲染的要求. 同时对于较为复杂的场景, 我们的方法可以有效提升画面的渲染速度. 例如, 在 1920×1080 高分辨率并启用光线追踪的条件下, Medieval Docks 和 Bunker 场景存在高光反射丰富与含较多半透明材质的特殊环境, 这些环境下画面帧的渲染时间达到了 141 ms, 尤其是 Medieval Docks 场景的每帧平均渲染时长接近 100 ms, 会在实时渲染场景下会产生严重的卡顿, 远低于同分辨率几何缓存加上低分辨率图像生成与网络推理的总时间开销 (约为 60 ms). 而 Redwood Forest 场景本身的光照计算量已经有较大优化, 因此直接渲染高清图像的开销较小. 值得注意的是, 我们的网络可以进一步优化性能, 例如, 使用 CUDA 和 cuDNN[42]优化, 而不是使用 TensorRT 下的 ONNX 网络推理. 这些优化方法的探索将作为我们未来的研究工作.

4.4 消融研究

在本节中, 我们对网络模型中的融合部分与作为网络输入的几何缓存进行了消融研究, 以评估它们在本次研究工作中的有效性.

4.4.1 融合部分

相较于简单的连接 (concat) 或相加 (add) 操作, 我们在网络模型中设计的融合模块会对几何缓存特征进行 SE 操作, 再让其与内容图像特征经过残差块以填补内容细节, 最终将其与经过 SE 操作的几何缓存连接. 为了证明该融合模块在网络结构中的有效性, 我们将网络中的各融合模块替换为连接与相加操作, 并对网络进行重新训练和测试, 对比结果如表 3 所示.

表 3 不同场景下各融合操作的性能指标对比. 超分规模均为 4×4

指标	场景	Add	Concat	Ours
PSNR (dB)	Bunker	27.86	28.00	28.42
	Redwood Forest	23.91	24.10	24.25
	Medieval Docks	25.14	25.32	25.37
SSIM	Bunker	0.8916	0.8959	0.9041
	Redwood Forest	0.8641	0.8675	0.8712
	Medieval Docks	0.8732	0.8683	0.8705

从表 3 中可以看出, 在不同场景下网络中的融合操作对应的性能指标总体优于简单的连接和相加操作, 而且相加操作的大部分性能指标明显低于其他两种操作, 这是由于相加操作的结果特征的通道数更少, 无法为网络提

供足够丰富的信息. 同时在主观感知层面我们比较了连接操作与网络中融合操作的预测结果, 对比示例如图 10 所示. 在图 10 的第 1 行示例中, 我们可以看到左边开关的下方存在高光反射, 将连接作为融合操作的模型无法将其进行还原, 使得红色的阀门开关呈现出了不自然的黄色, 将相加作为融合操作的模型能进行一定程度的还原, 但亮度偏低. 类似的, 第 2 行示例中连接和相加结果预测草叶会有不自然的高光, 而第 3 行示例中树木产生的阴影区域更为广泛, 而连接操作和相加操作产生的阴影较小. 这两行示例反映出简单的连接或相加操作无法有效区分几何缓存与内容图像的特征作用, 从而过于依赖几何缓存的细节特征, 但几何缓存中不含阴影, 高光等这些信息, 因此其产生的结果在这些部分与真值内容差异较大. 而我们的融合操作对几何缓存特征进行注意力区分, 并利用其对内容图像特征中的无效信息进行定位和修正, 从而预测得到更为合理的超分结果.

(a) Add (b) Concat (c) Ours (d) GT

图 10 不同场景下各融合操作的视觉结果对比

4.4.2 几何缓存种类

为了研究在模型学习过程中提供的几何缓存的作用, 我们在网络输入分别移除了不同的几何缓存 (几何缓存种类见第 3.1 节), 并相应调整了几何缓存对应“浅层特征提取”部分中最后一层卷积的输出通道数, 使几何缓存与内容图像的特征通道数仍相等. 我们在 Bunker 数据集上重新训练和测试了网络模型, 测试产生的性能指标对比见表 4. 其中, –bc, –wn, –sd, –r, –m 分别表示移除底色, 法向, 场景深度, 粗糙度和金属色. 超分规模均为 4×4.

表 4 Bunker 场景下移除各几何缓存的性能指标对比

指标	–bc	–wn	–sd	–m	–r	ours
PSNR (dB)	27.91	27.43	28.01	28.19	28.13	28.42
SSIM	0.8778	0.8748	0.8974	0.9013	0.8948	0.9041

从表 4 的数据结果中可以看出, 我们的网络模型将包含底色, 法向, 场景深度, 金属色, 粗糙度的几何缓存作为输入, 作为训练时损失函数一部分的 SSIM 指标达到了 0.9041. 而在移除场景深度后, 指标下降到 0.8974. 类似地, 移除单通道的粗糙度或金属色后, PSNR 指标均下降了 0.2 dB 左右. 移除底色与法向的模型在 SSIM 的度量指标下降较多, 这是由于这两种几何缓存均为三通道数, 移除后减少的信息量多于其他一通道数的几何缓存. 同时移除法向的模型在 PSNR 和 SSIM 指标的结果数值上下降最多, 对此一种合理的解释是, 一方面是法向本身存储的信息会影响光线在物体表面反射与折射的结果, 从而在学习过程中移除法向会导致模型对高光反射等一些区域缺乏有效信息并预测出较差的超分结果. 另一方面在于屏幕空间下法向的像素间变化明显, 更容易检测出边缘与角点,

这使得图像重建过程能获得更多的高频特征, 对高清内容图像的细节内容进行合理复原, 提升了超分结果的保真度.

另外地, 为了进一步验证本文所提的网络架构相对于以往网络的有效性, 我们将所使用的几何缓存进行了删减, 只保留场景深度, 从而与 NSRR 中所使用的几何缓存保持一致. 表 5 给出了场景深度的消融实验结果, 其中+sd 表示作为输入的几何缓存仅保留场景深度, 超分规模均为 4×4, 从表 5 的指标对比上可以看出, 该模型表现优于第 4.2 节中 NSRR[9]的方法, 说明即使在不使用历史信息的情况下, 我们的方法仍可以用多层次融合机制等特点获得更优秀的画面表现, 而且使用更多的几何缓存信息可以进一步提升超分画面质量.

表 5 Medieval Docks 场景下移除场景深度的性能指标对比

指标	+sd	NSRR	Ours
PSNR (dB)	22.40	20.79	25.37
SSIM	0.730 5	0.700 6	0.870 5

4.4.3 整体网络结构

与传统的 U-Net[21]结构不同, 我们的模型设计考虑对渲染图像与几何缓存使用各自分支的编码器, 这有助于网络从不同分布的信息中提取有效特征, 而跳跃连接的基础上引入的融合机制则提高了网络的表达能力. 我们将网络结构替换为原始的 U-Net[21]结构后重新进行训练和测试, 得到如图 11 所示的预测结果. 从图中箭头所指区域可以看出, 阴影区域存在不连续与缺失的问题, 这说明 U-Net 结构的重建能力差于我们的模型, 其原因在于单分支的编码结构与缺少融合模块使得网络无法更好地提取与识别有效信息, 从而在包含阴影等复杂信息的区域会存在预测错误.

(a) Input (b) U-Net (c) Ours (d) GT

图 11 U-Net 结构与我们模型的预测结果对比

5 结 论

5.1 局限性

由于几何缓存的局限性以及低清输入丢失信息过多等原因, 我们的方法对一些包含阴影和高光反射的区域进行超分时, 会恢复出与真值图像不相符的结果, 图 12 展示了网络在超分预测时的错误案例. 从图 12 可以看出, 红色阀门的倒影在墙壁上本应形成近似椭圆的形状, 但输入丢失了这部分信息, 而我们使用的几何缓存中并不包含光源位置、照射方向等信息, 导致网络无法对该区域内的阴影进行有效补全, 从而产生了不合理的结果.

为了保证图片的质量, 我们目前选择对每个场景训练一个独立的模型, 而能适应多个场景的泛化模型, 囿于模型训练与测试所需的巨大开销, 例如泛化所需庞大的场景数据资源, 本次工作并未进行泛化模型的相关实验与讨论, 我们将其作为未来研究与实践的一个话题.

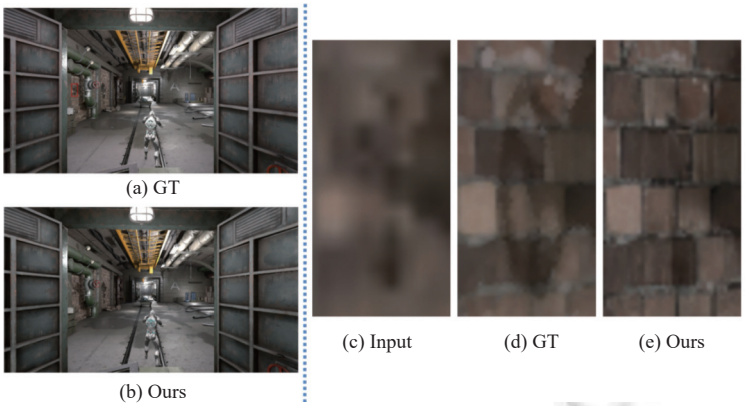


图 12 网络模型预测错误的示例

5.2 总结与展望

本文提出了一种对渲染图像进行超分的新方法. 该方法不依赖历史帧和相关运动信息, 而是利用渲染器的高分辨率的几何缓存来帮助填充上采样过程中的细节信息. 我们在网络结构中引入了一种新的特征融合机制, 在重建网络中使用层次化的融合模块来增强几何缓存特征对图像特征的内容补充效果. 同时大量实验结果证明, 对比于其他现有的图像和视频超分方法, 我们的工作能在不同场景下都展现出更好的超分结果. 相比于传统的渲染器工作流程, 在一些需要大量计算的场景下使用我们的超分方法辅助渲染帧, 可以在保证渲染内容画质的前提下提高渲染效率, 增强画面流畅性.

本文提出的研究方法在运行时间上仍存在一定的优化空间, 可以通过 cuDNN[42]等硬件优化与网络量化等软件工程方法来改进模型结构与运行过程, 从而更好地提升网络性能. 并且我们的工作基于空间上的分辨率大小进行渲染过程改良, 可以与诸如帧内外插值等技术相结合, 在时间和空间上对渲染内容进行超采样研究, 进一步提升渲染环境的运行帧率. 部署在手机等移动设备上的渲染管线通常与电脑端有着不同的架构与工作流程, 因此将当前工作推广到不同设备的渲染平台与相关的优化手段是我们之后需要研究的方向. 提升不同场景间模型学习的泛化性同样是一个有趣的未来话题. 我们相信, 在我们的方法中提出的融合策略与高质量结果能为之后的渲染图像超分工作提供更多帮助与借鉴.

References:

[1] Kaplanyan AS, Sochenov A, Leimkühler T, Okunev M, Goodall T, Rufo G. DeepFovea: Neural reconstruction for foveated rendering and video compression using learned statistics of natural videos. ACM Trans. on Graphics, 2019, 38(6): 212. [doi: 10.1145/3355089.3356557]

[2] Jo Y, Oh SW, Kang J, Kim SJ. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 3224–3232. [doi: 10.1109/CVPR.2018.00340]

[3] Dong C, Loy CC, Tang XO. Accelerating the super-resolution convolutional neural network. In: Proc. of the 14th European Conf. on Computer Vision. Amsterdam: Springer, 2016. 391–407. [doi: 10.1007/978-3-319-46475-6_25]

[4] Epic Games. 2022. https://www.unrealengine.com/.UnrealEngine

[5] Burnes A. NVIDIA DLSS 2.0: A big leap in AI rendering. 2020. https://www.nvidia.com/en-us/geforce/news/nvidia-dlss-2-0-a-big-leap-in-ai-rendering/.NVIDIA

[6] DirectX-Specs, Variable Rate Shading. 2022. https://microsoft.github.io/DirectX-Specs/d3d/VariableRateShading.html.Microsoft

[7] Xiao K, Liktor G, Vaidyanathan K. Coarse pixel shading with temporal supersampling. In: Proc. of the 2018 ACM SIGGRAPH Symp. on Interactive 3D Graphics and Games. Montreal: ACM, 2018. [doi: 10.1145/3190834.3190850]

[8] Sakai H, Nabata K, Yasuaki S, Iwasaki K. Error estimation for many-light rendering with supersampling. In: Proc. of the 2018 SIGGRAPH Asia Technical Briefs. Tokyo: ACM, 2018. [doi: 10.1145/3283254.3283264]

[9] Xiao L, Nouri S, Chapman M, Fix A, Lanman D, Kaplanyan A. Neural supersampling for real-time rendering. ACM Trans. on Graphics, 2020, 39(4): 142:1–142:12. [doi: 10.1145/3386569.3392376]

[10] Kalantari NK, Bako S, Sen P. A machine learning approach for filtering Monte Carlo noise. ACM Trans. on Graphics, 2015, 34(4):

122:1–122:12. [doi: 10.1145/2766977]

[11] Nalbach O, Arabadzhiyska E, Mehta D, Seidel HP, Ritschel T. Deep shading: Convolutional neural networks for screen space shading. Computer Graphics Forum, 2017, 36(4): 65–78. [doi: 10.1111/cgf.13225]

[12] Isobe T, Zhu F, Jia X, Wang SJ. Revisiting temporal modeling for video super-resolution. arXiv:2008.05765, 2020.

[13] Haris M, Shakhnarovich G, Ukita N. Space-time-aware multi-resolution video enhancement. In: Proc. of the 2020 IEEE/CVF Conf. on Computer Vision and Pattern Recognition (CVPR). Seattle: IEEE, 2020. 2856–2865. [doi: 10.1109/CVPR42600.2020.00293]

[14] Liao RJ, Tao X, Li RY, Ma ZY, Jia JY. Video super-resolution via deep draft-ensemble learning. In: Proc. of the 2015 IEEE Int'l Conf. on Computer Vision. Santiago: IEEE, 2015. 531–539. [doi: 10.1109/ICCV.2015.68]

[15] Chan KCK, Zhou SC, Xu XY, Loy CC. BasicVSR++: Improving video super-resolution with enhanced propagation and alignment. In: Proc. of the 2022 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. New Orleans: IEEE, 2022. 5962–5971. [doi: 10.1109/CVPR52688.2022.00588]

[16] Dong C, Loy CC, He KM, Tang XO. Learning a deep convolutional network for image super-resolution. In: Proc. of the 13th European Conf. on Computer Vision. Zurich: Springer, 2014. 184–199. [doi: 10.1007/978-3-319-10593-2_13]

[17] Lim B, Son S, Kim H, Nah S, Lee KM. Enhanced deep residual networks for single image super-resolution. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition Workshops. Honolulu: IEEE. 2017. 1132–1140. [doi: 10.1109/CVPRW.2017.151]

[18] Zhang YL, Li KP, Li K, Wang LC, Zhong BN, Fu, Y. Image super-resolution using very deep residual channel attention networks. In: Proc. of the 15th European Conf. on Computer Vision (ECCV). Munich: Springer, 2018. 294–310. [doi: 10.1007/978-3-030-01234-2_18]

[19] Ledig C, Theis L, Huszár F, Caballero J, Cunningham A, Acosta A, AitkenA, Tejani A, Totz J, Wang ZH and Shi WZ. Photo-realistic single image super-resolution using a generative adversarial network. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 105–114. [doi: 10.1109/CVPR.2017.19]

[20] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556, 2014.

[21] Weng WH, Zhu X. INet: Convolutional networks for biomedical image segmentation. IEEE Access, 2021, 9: 16591–166603. [doi: 10.1109/ACCESS.2021.3053408]

[22] Ding L, Ding SF, Zhang J, Zhang ZC. Single image super-resolution reconstruction based on VGG energy loss. Ruan Jian Xue Bao/Journal of Software, 2021, 32(11): 3659–3668. (in Chinese with English abstract) http://www.jos.org.cn/1000-9825/6053.htm [doi: 10.13328/j.cnki.jos.006053]

[23] Pan ZX, Yu J, Xiao CB, Sun WD. Single image super resolution based on adaptive multi-dictionary learning. Acta Electronica Sinica, 2015, 43(2): 209–216 (in Chinese with English abstract). [doi: 10.3969/j.issn.0372-2112.2015.02.001]

[24] Caballero J, Ledig C, Aitken A, Acosta A, Totz J, Wang ZH, Shi WZ. Real-time video super-resolution with spatio-temporal networks and motion compensation. In: Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition. Honolulu: IEEE, 2017. 2848–2857. [doi: 10.1109/CVPR.2017.304]

[25] Wang XT, Chan KCK, Yu K, Dong C, Loy CC. EDVR: Video restoration with enhanced deformable convolutional networks. In: Proc. of the 2019 IEEE/CVF Conf. on Computer Vision and Pattern Recognition Workshops (CVPRW). Long Beach: IEEE, 2019. 1954–1963. [doi: 10.1109/CVPRW.2019.00247]

[26] Yi P, Wang ZY, Jiang K, Jiang JJ, Ma JY. Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In: Proc. of the 2019 IEEE/CVF Int'l Conf. on Computer Vision. Seoul: IEEE, 2019. 3106–3115. [doi: 10.1109/ICCV.2019.00320]

[27] Lu YF, Xie N, Shen HT. DMCR-GAN: Adversarial denoising for Monte Carlo renderings with residual attention networks and hierarchical features modulation of auxiliary buffers. In: SIGGRAPH Asia 2020 Technical Communications. ACM, 2020. 5. [doi: 10.1145/3410700.3425426]

[28] Chaitanya CRA, Kaplanyan AS, Schied C, Salvi M, Lefohn A, Nowrouzezahrai D, Aila T. Interactive reconstruction of Monte Carlo image sequences using a recurrent denoising autoencoder. ACM Trans. on Graphics, 2017, 36(4): 98:1–98:12. [doi: 10.1145/3072959.3073601]

[29] Guo J, Fu XH, Lin LQ, Ma HJ, Guo YW, Liu SQ, Yan LQ. ExtraNet: Real-time extrapolated rendering for low-latency temporal supersampling. ACM Trans. on Graphics, 2021, 40(6): 278:1–278:16. [doi: 10.1145/3478513.3480531]

[30] Du WJ, Feng JQ. A unified anti-aliasing method for deferred shading. Journal of Computer-aided Design & Computer Graphics, 2016, 28(1): 58–67 (in Chinese with English abstract). [doi: 10.3969/j.issn.1003-9775.2016.01.008]

[31] Shao P, Zhou W, Li GQ, Wu ZJ. Improved anti-aliasing algorithm based on deferred shading. Computer Science, 2018, 45(11A): 218–221, 225 (in Chinese with English abstract).

[32] Briedis KM, Djelouah A, Meyer M, McGonigal I, Gross M, Schroers C. Neural frame interpolation for rendered content. ACM Trans. on

Graphics, 2021, 40(6): 239. [doi: 10.1145/3478513.3480553]
[33] Hu J, Shen L, Sun G. Squeeze-and-excitation networks. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 7132–7141. [doi: 10.1109/CVPR.2018.00745]
[34] He KM, Zhang XY, Ren SQ, Sun, J. Deep residual learning for image recognition. In: Proc. of the 2016 IEEE Conf. on Computer Vision and Pattern Recognition. Las Vegas: IEEE, 2016. 770–778. [doi: 10.1109/CVPR.2016.90]
[35] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP. Image quality assessment: From error visibility to structural similarity. IEEE Trans. on Image Processing, 2004, 13(4): 600–612. [doi: 10.1109/TIP.2003.819861]
[36] Zhang R, Isola P, Efros AA, Shechtman E, Wang O. The unreasonable effectiveness of deep features as a perceptual metric. In: Proc. of the 2018 IEEE/CVF Conf. on Computer Vision and Pattern Recognition. Salt Lake City: IEEE, 2018. 586–595. [doi: 10.1109/CVPR.2018.00068]
[37] Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen J, Lin ZM, Gimelshein N, Antiga L, Desmaison A, Köpf A, Yang E, DeVito Z, Raison M, Tejani A, Chilamkurthy S, Steiner B, Fang L, Bai JJ, Chintala, S. PyTorch: An imperative style, high-performance deep learning library. In: Proc. of the 33rd Int'l Conf. on Neural Information Processing Systems. Vancouver: Curran Associates Inc., 2019. 721.
[38] Kingma DP, Ba J. Adam: A method for stochastic optimization. In: Proc. of the 3rd Int'l Conf. on Learning Representations. San Diego: ICLR, 2015.
[39] Glorot X, Bengio Y. Understanding the difficulty of training deep feedforward neural networks. In: Proc. of the 13th Int'l Conf. on Artificial Intelligence and Statistics. Sardinia: JMLR.org, 2010. 249–256.
[40] Open Neural Network Exchange. 2022. https://onnx.ai/.ONNX
[41] NVIDIA TensorRT. 2022. https://github.com/NVIDIA/TensorRT.NVIDIA
[42] NVIDIA. 2022. https://developer.nvidia.com/cudnn.NVIDIAcuDNN

附中文参考文献:

[22] 丁玲, 丁世飞, 张健, 张子晨. 使用VGG能量损失的单图像超分辨率重建. 软件学报, 2021, 32(11): 3659–3668. http://www.jos.org.cn/1000-9825/6053.htm [doi: 10.13328/j.cnki.jos.006053]
[23] 潘宗序, 禹晶, 肖创柏, 孙卫东. 基于自适应多字典学习的单幅图像超分辨率算法. 电子学报, 2015, 43(2): 209–216. [doi: 10.3969/j.issn.0372-2112.2015.02.001]
[30] 杜文俊, 冯结青. 面向延迟着色的统一反走样算法. 计算机辅助设计与图形学学报, 2016, 28(1): 58–67. [doi: 10.3969/j.issn.1003-9775.2016.01.008]
[31] 邵鹏, 周伟, 李光泉, 吴志健. 一种后处理式的改进抗锯齿算法. 计算机科学, 2018, 45(11A): 218–221, 225.

张浩南(2000－), 男, 硕士生, 主要研究领域为计算机图形学.

傅锡豪(1997－), 男, 硕士生, 主要研究领域为实时渲染.

过洁(1986－), 男, 博士, 副研究员, CCF 高级会员, 主要研究领域为计算机图形学, 虚拟现实.

郭延文(1980－), 男, 博士, 教授, 博士生导师, CCF 专业会员, 主要研究领域为计算机图形学, 三维视觉.

覃浩宇(1998－), 男, 硕士生, 主要研究领域为计算机图形学.