基于神经网络的超分辨率实时渲染

吴雨桐，余北辰

摘 要: 随着游戏行业的发展，人们对电子游戏画面的分辨率要求越来越高，游戏引擎和相关硬件如何实时地将低分辨率的游戏画面超分辨渲染到高分辨率成为一个重要问题。为同时兼顾超分辨率渲染的实时性和保真度，本课题组参考facebook 团队的论文，Neural supersampling for real-time rendering[1] ，设计了基于神经网络的超分辨率实时渲染算法，实现了保证保真度的同时，能够在极短时间内（实时）完成渲染。

Title: Real-time super-resolution rendering based on neural network

Yutong Wu, Beichen Yu

**Abstract**:

**With the development of the game industry, people have higher and higher requirements for the resolution of video game screens. How the game engine and related hardware can super-resolution rendering of low-resolution game screens to high-resolution in real time has become an important issue. In order to take into account the real-time and fidelity of super-resolution rendering at the same time, our team referred to the facebook team’s paper, Neural supersampling for real-time rendering[1] , and designed a neural network-based super-resolution real-time rendering algorithm to achieve guaranteed fidelity at real-time.**

**Key word：deep learning, rendering, upsampling, superresolution**

# 简介与意义/Introduction

## 项目意义和依据/Significance

实时渲染技术（real-time rendering）是电子游戏和VR设备的一项关键技术，在目前的游戏引擎中广泛应用，其目的在于能在更短的时间内对图片进行渲染。并且随着屏幕分辨率和刷新率的不断提升，以及对逼真效果的更高要求，实时渲染技术作为一项关键技术，也面临很多挑战，也亟需得到改进。

现有游戏引擎实时渲染的做法一般是先对低分辨率进行渲染，然后通过上采样（upsample）还原到高分辨率的原图。目前为止，已经提出了较多的方法，比如Oculus Quest的foveated rendering、Unreal Engine提出的TAAU以及Nvidia发布的DLSS。但这些已有的方法都有共同的缺陷：要么会在上采样后的图片引入明显的视觉差错（特别是当上采样频率高于2×2时），要么会依赖于一些专有的技术和硬件，这些技术和硬件不是在所有平台上都能获得并使用的。

本课题组的工作是基于Facebook团队提出的神经网络超分辨率实时渲染的方法[1] ，搭建神经网络模型，并自己生成数据集并完成机器学习的训练，然后与现有的超分辨率方法进行运行效率和渲染画质的比较。本课题组实现的方法的优点在于此方法很容易与现有的游戏引擎集成，而且不需要特殊的硬件设备。这使得该方法能应用于广泛的现有软件平台。再者，此方法的输入采用常见的游戏引擎的输入（低分辨率的色彩、深度和运动矢量），实现了使用4×4的超采样频率（现有方法超采样频率一般不超过2×2）对严重走样的输

入进行渲染，并且产生高度逼真和时间稳定的结果。

## 本方法/系统框架/Article Structure

本课题组的工作是基于Facebook Reality Labs的Neural Supersampling for Real-time Rendering[1] ，原文中是使用前4帧的信息对当前帧进行超分辨率，本课题组在比较使用前帧的数量对效果的影响之后，选择使用前5帧来超分辨率当前帧。本方法的框架如下图 1 系统框架所示：

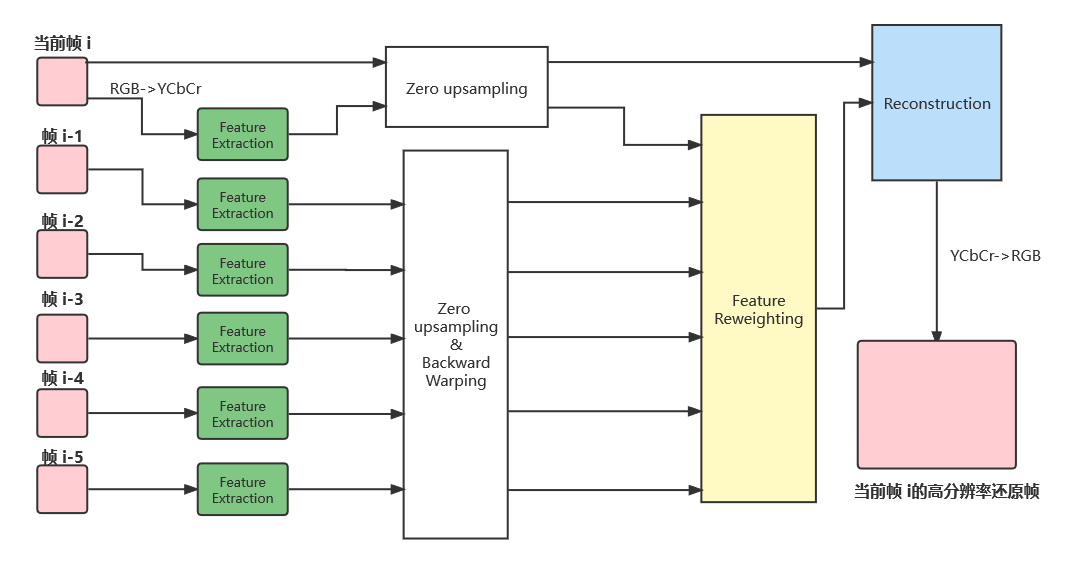


图 1 系统框架

Feature Extraction、Feature Rweighting以及Reconstruction部分为三个子神经网络，Zero upsampling、Back warping为两个对图像的基本操作（不涉及神经网络）。其流程解释如下：

1. 对于当前帧，有两条流程路径，第一条：首先利用Feature Extraion模块对帧进行特征提取，得到的输出为12个通道的特征（包括颜色，深度以及8个学习得到的特征），再zero upsampling之后作为Reconstruction的输入，作为最后重构帧的基础；第二条：直接zero upsampling在之后作为Feature Rweighting的输入，用于计算前4帧的权重。
2. 对于前4帧，其目的在于运用运动矢量（motion vector）的连续性来实现对当前帧的高清重构，它们的处理流程一致，先进性特征提取，在通过zero upsampling和backward warping实现时间重投影，其结果作为Feature Rweighting的输入，计算出每一帧对应的权重矩阵，这个矩阵与对应帧的全部12个特征相乘，结果作为重构模块的输入。
3. 用Reconstruction模块把当前帧的零上采样和获得的前帧的权重合成出目标结果（当前帧的高分辨率的渲染图片）。

# 相关工作/Related Works

本工作的相关工作主要是解决实时渲染中的采样问题以及图片/视频超分辨率重构问题。

## 实时渲染

实时渲染的一个重要问题是采样频率不足，这将导致走样问题，在本工作的相关方法中，解决采样频率不足的方法可以划分为两类：空间反走样（Spatial Antialasing）和 时间反走样（temporal Antialasing）。

### 空间反走样

大多是空间反走样的主要思路是对图像的细节进行模糊处理，从而使锯齿状边缘更加平滑。经典方法是MSAA [Akeley 1993][2] ，其做法是将像素划分为更小的像素单元，从而增大的采样频率；还有更新的方法，比如 MLAA [Reshetov 2009][3] ，其做法是用得到的图片上颜色的间断性特征来估计原图中像素的覆盖率；SMAA [Jimenez et al. 2012][4] ，此方法将MSAA与MLAA结合，得到更好的效果。

### 时间反走样

深度学习超采样 (DLSS) [Edelsten et al. 2019][5] 是与本工作最相似的方法，它使用时间历史和神经网络来增强边缘并执行放大。最近降低渲染成本的另一个趋势是将重建方法应用于sparsely ray-traced [Schied et al. 2017][6] 和foveated images [Patney et al. 2016][7] 。下述工作与本课题组的上采样插值有关：real-time low-sample-count reconstruction [Chaitanya et al. 2017] [8] foveated reconstruction [Kaplanyan et al. 2019][9] 。这些方法训练得到时间稳定的 U-Net 架构，实现从非常嘈杂或者稀疏的输入帧中获得稳定的重建帧。

## 图片/视频超分辨率

### 图片超分辨率

单个图片进行超分辨率（SISR）即从一个低分辨率图片还原出其高分辨率结果，目前相关工作主要是通过卷积神经网络和残差模块的应用：比如较早的是 SRCNN [Dong et al. 2015][10] 就是应用了一个三层CNN；之后从 VDSR [Kim et al. 2016][11] ，开始对两张图（目标图和原图）的残差进行学习，包括后面的EDSR [Lim et al. 2017][12] 也是类似的方法。ESPCN [Shi et al. 2016][13] 介绍了一种以低分辨率运行并实现实时性能的子像素的CNN。 LapSRN [Lai et al. 2017][14] 提出了一个拉普拉斯金字塔网络，用于逐步重建高分辨率图像的子带残差。 RDN [Zhang et al. 2018b][15] 结合残差和dense connections [Huang et al. 2017][16] 用于分层特征提取。 RCAN [Zhang et al. 2018a][17] 引入了一个残差的的残差（residual-in-residual）结构来形成一个非常深的网络，即超过 400 个卷积层，实现最先进的质量结果。

### 视频超分辨率

视频超分辨率方法通常利用相邻帧中的时间相干性来改进 SISR 方法的重建。大多数方法的一个关键组成部分是帧之间的运动估计。

VESPCN [Caballero et al. 2017][18] 引入了 multial-resolution spatial transformer 模块用于运动补偿和视频超分辨率。EDVR [Wang et al. 2019][19] 应用金字塔、级联和可变形对齐模块以及时间和空间注意模块。

还有的方法是使用递归神经网络 (RNN)。 FRVSR [Sajjadi et al. 2018][20] 提出了一种 RNN，可以扭曲（warping）先前估计的帧来还原后续帧。TecoGAN [Chu et al. 2018][21] 通过改善训练损失和引入时间上的自我监督对抗学习算法来解决了RNN 由于训练时采用简单的范数损失函数，导致它们的结果可能缺乏空间细节的问题。

上述相关工作都几乎有一个共同特点，就是需要特殊（高性能）的渲染硬件以及所需时间都较长，难以满足实时性的要求（特别是对上采样频率高于2×2时，比如本课题组实验用的上采样频率为4×4），而本课题组所实现的算法能够在高上采样率的情况下，实时渲染出高保真度的画面。

# 研究内容与方法(或算法)/Contnts and Methods(or Algorithm)

## 总体流程

本课题组主要通过基于神经网络的深度学习超采样方法，针对输入的低分辨率的视频片段进行逐帧的实时采样，通过深度学习的方法还原出高分辨率的信息，从而输出所对应的高分辨率的视频片段。与现有方法

主要进行2×2超分辨率采样相比，本方法能够实现低延时的4×4超分辨率采样。

本方法的流程主要由四个子模块所组成：Feature Extraction 特征提取，Temporal Reprojection 时序重投影，Feature Reweighting 特征重加权，Reconstruction 重建。输入的低分辨率的视频片段分别经过四个子模块的处理后，即可输出对应的还原至高分辨率的结果。过程流程图如图 2 所示：

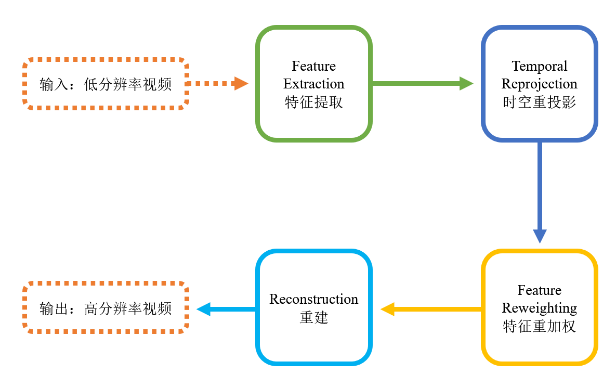


图 2 子模块流程图

## 特征提取（Feature Extraction）模块

首先，对于输入的低分辨率视频，需要通过**特征提取**模块的子网络，解析每一帧的特征。该模块采用三层卷积神经网络（CNN），独立对每一个输入的帧进行处理。此时输入的帧位于低分辨率空间下。对于每一帧而言，输入该帧的RGB-D信息进入神经网络，通过网络生成8通道的学习特征。与输入的RGB-D信息相链接后，特征提取模块将输出12通道的特征信息作为后续的输入。由于在YCbCr空间上进行超采样的效果更佳，在实际操作过程中，本模块将当前帧的颜色信息从RGB空间先转换至YCbCr空间，再输入给子网络进行特征提取。

特征提取模块子网络的结构示意图如图 3 所示：

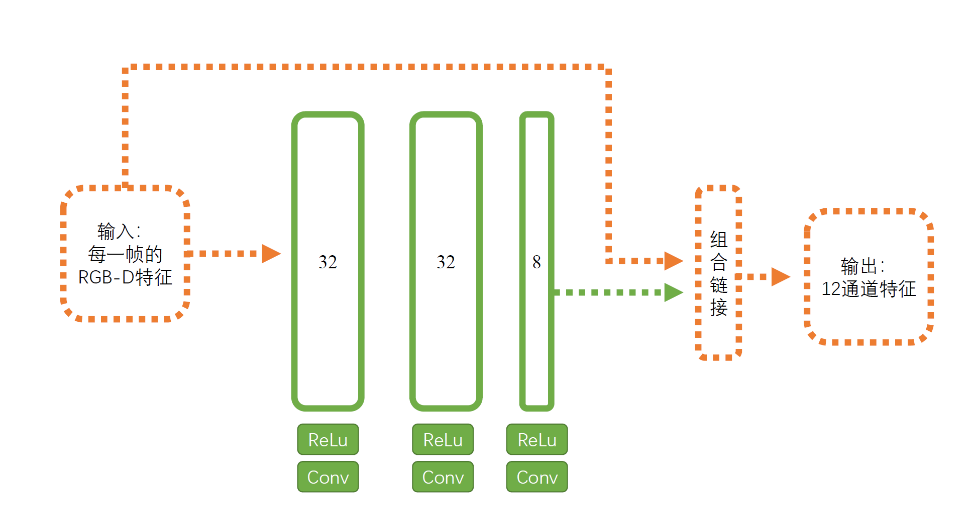


图 3 特征提取模块子网络示意图

## 时序重投影（Temporal Reprojection）模块

流程的下一个模块是进行**时序重投影**。进行时序重投影的目的是减少重建网络所需的感受野，从而减少重建网络的复杂性。时序重投影模块的操作对象限于先前帧，通过运动矢量将先前帧所学习的特征投影到当前帧。为了更好地利用渲染数据的细节，本方法选择先将运动矢量映射至高分辨率空间，再进行重投影（这是由于运动矢量在物理上更具有连续性，在后续对运动矢量双线性插值时效果较好）。时序重投影的过程主要分为以下四步：

第一步：对每一个输入的像素进行从低分辨率空间到高分辨率空间的零上采样，即直接将低分辨率空间的像素映射到高分辨率空间对应的位置上，并将其他位置的像素置零；

第二步：对于运动矢量，使用双线性插值上采样的方法，将其从低分辨率空间同样映射到高分辨率空间；

第三步：将前5帧的运动矢量依次迭代warp到当前帧。更加通俗的来说，例如当前帧为第i帧，则对于得到零上采样的第i-5帧的运动矢量，需要依次和第i-4，i-3，i-2，i-1帧的运动矢量进行warp，从而得到第i-5帧到第i帧（当前帧）的运动矢量，这是为了后续辅助rdb和depth等特征warp到当前帧。

第四步：在高分辨率空间上，每一个前帧对于除了运动矢量以外的特征，应用第三步中得到的该帧到当前帧的运动矢量进行向后翘曲（backward warping），使其与当前帧对齐。

此模块是本课题组所复现论文对于超分辨率中抗锯齿和插值问题的创新解决方案，其核心是利用率运动矢量在物理客观上的连续性。显然，在进行时序重投影时，选择越多的先前帧，将会收获越好的输出效果，但相应消耗的资源代价也会越大。因此，在本课题组的实验中，选择采样前5帧来进行时序重投影。

## 特征重加权（Feature Reweighting）模块

经历时序重投影后的运动矢量不能完全正确反映出帧与帧之间发生遮挡或者产生阴影的情况，因此经过翘曲后的帧很有可能会包含伪影。因此需要使用一个**特征重加权**的模块对此进行加权处理。对每一个所采样的先前帧，该模块计算出一个权重矩阵，再将所有帧的特征带权累加输出重加权后的结果，以屏蔽可能出现的伪影。该模块同样采用三层卷积神经网络（CNN）进行处理，将所有的先前帧经过零上采样的翘曲特征和当前帧的RGB-D信息打包输入。神经网络对于每一个先前帧逐像素预测权重，再用预测的权重乘以每一对应先前帧的所有特征（12通道），作为神经网络的输出结果。

特征重加权模块子网络的结构示意图如图 4 所示：

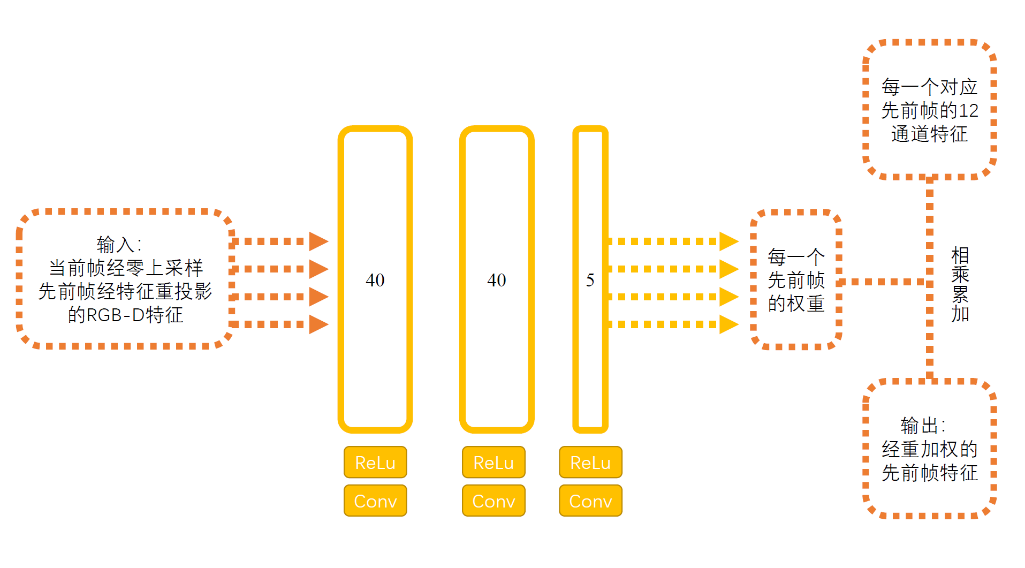


图 4 特征重加权模块子网络示意图

这个网络结构与原论文中的不同，因为本课题组采取的是前5帧还原当前帧，所以经过3层卷积神经网络之后的输出维度为5（5个前帧分别对应一个矩阵），所以按照原论文的比例，依次扩大了每一层卷积的维度，如图 4所示。

## 重建（Reconstruction）模块

经历了上述模块后，通过**重建**网络重建出最终的高分辨率输出结果。重建网络使用3级、10层的U-Net，输入当前帧的特征和之前得到的经过重新加权的先前帧的特征，经过神经网络，输出经过重建后的高分辨率图像作为输出结果。

重建模块子网络的结构示意图如图 5 所示：

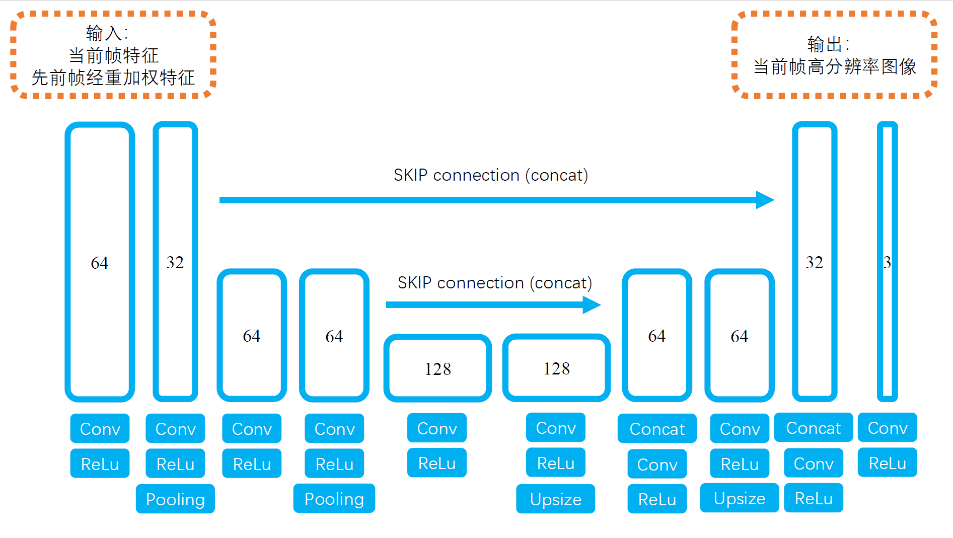


图 5 特征重加权模块子网络示意图

## 损失函数的设置

在训练网络的过程中需要使用损失函数作为准则，对训练的效果进行评估。在本方法中，采用以下函数作为整个网络的损失函数：



本方法中的损失函数采用的是SSIM[Wang et al. 2004][22] 和预训练的VGG-16网络[Johnson et al. 2016][23] 的加权线性组合刻画的。公式中，是整个网络的输出结果，而是所选择的参考工作的输出结果。

# 实验结果与分析/Experiment Results and Analysis

本课题组使用unity生成了三个场景的动画，分别对2×2和4×4两种超采样率训练对应模型，训练过程，由于时间关系，每个场景采用了48帧，共192帧，输入到模型中训练了200个epoch。4×4超采样模型训练过程的loss如 图 6所示：

图 6 4×4超采样网络训练过程loss

本课题组选取FSRCNN，SRCNN，SRDenseNet，RDN这四种已有的基于深度学习的超分辨率算法作为参考，与本课题组的方法从渲染效果、运行效率等方面进行对比。在下文中，将对这四种方法的原理和网络结构逐一进行简单的介绍：

SRCNN[24] ：是学术界中第一个将深度学习的方法运用在超分辨率上的算法。SRCNN首先采用双三次插值（bicubic），将低分辨率的图像放大，再通过三层卷积网络（CNN）来对非线性映射进行拟合，最后输出高分辨率的结果。

FSRCNN[25] ：是在SRCNN的基础上进行的改进工作。与SRCNN相比，FSRCNN不需要另外将低分辨率图像放大，而是直接把低分辨率图像输入到网络，通过在网络中加入反卷积层来放大尺寸；FSRCNN在网络中使用更小的卷积核、使用更多的映射层，与SRCNN相比在速度上有较大的提升。

SRDenseNet[26] ：将稠密块的结构应用到了超分辨率问题上。主要分为四部分：首先，用一个卷积层学习低层的特征；接着，用多个稠密块学习高层的特征；然后，使用多个反卷积层学习上采样滤波器参数；最后，通过一个卷积层，生成高分辨率输出。

RDN[27] ：使用残余密集网络解决超分辨率问题。主要包含四部分：首先，使用浅层特征提取网络提取浅层特征；其次，使用残差密集块（RDBs）进行局部特征融合与残差连接；接着，使用密集特征融合（DFF）进行全局特征融合和残差连接；最后，使用上采样网络（UPNet）进行最后的上采样以及卷积操作。

## 效果展示

由于本课题组实现的算法相较于其他现有算法而言，突出优点在于对高倍数超采样的抗锯齿和插值，所以下面展示三个场景在4×4超分辨率时的测试情况和其他方法的对比以及一个场景的2×2超分辨率对比。如图 7、图 8、图 9所示。

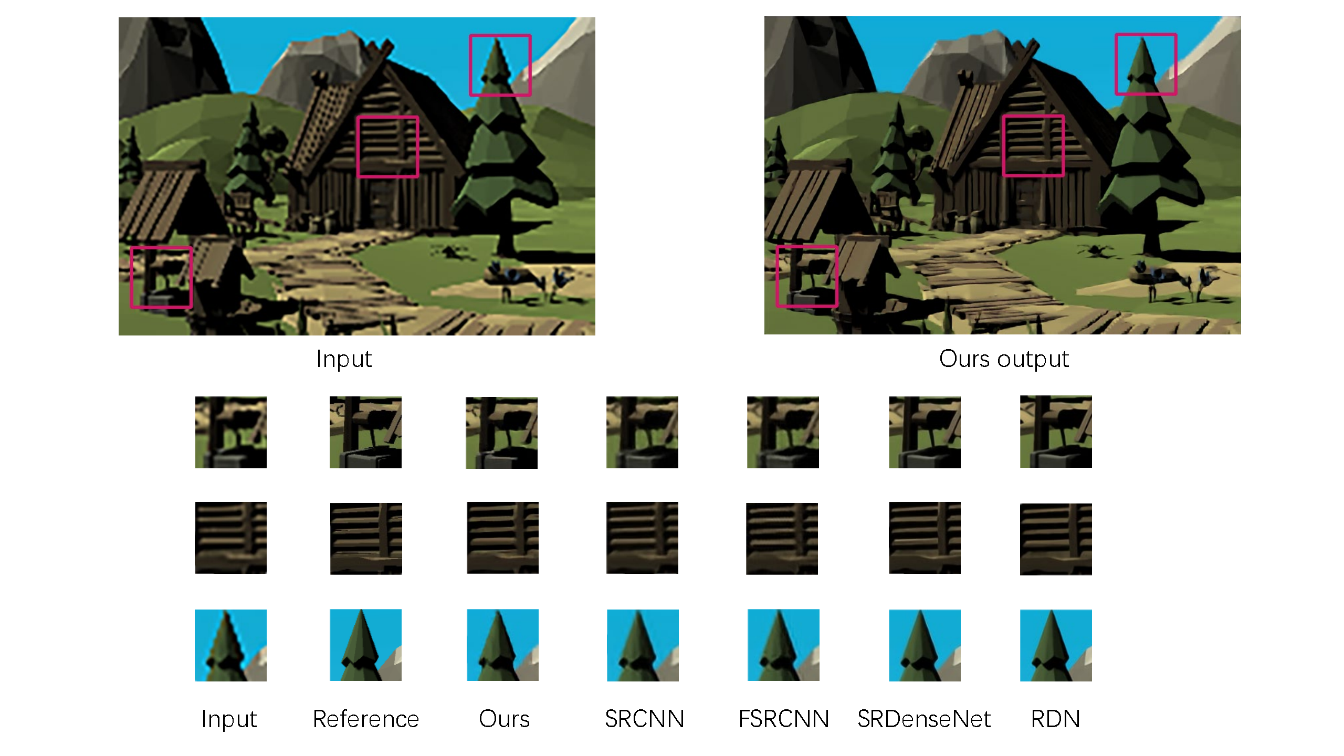
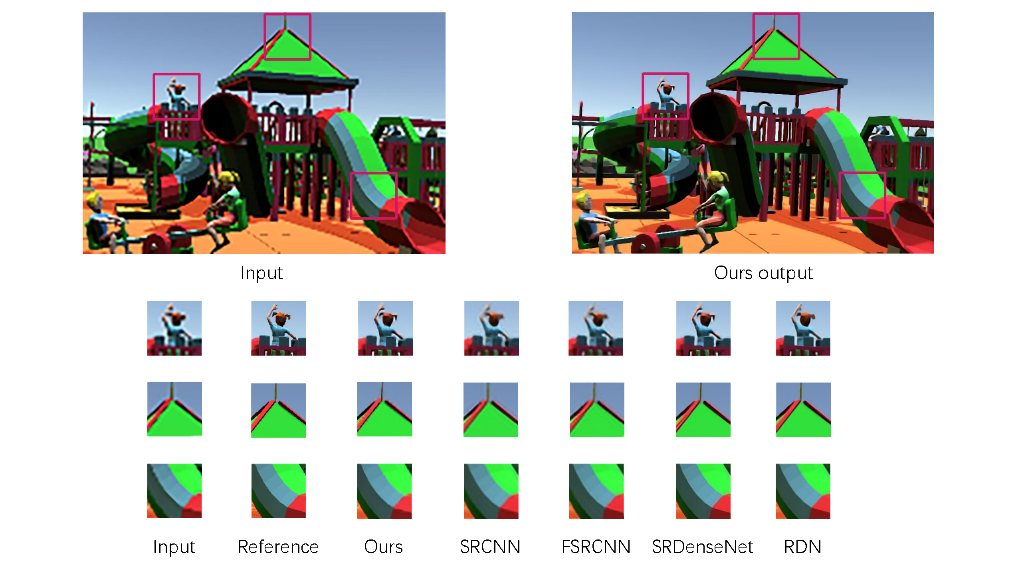


图 8 游乐园场景4×4超分辨率对比图

图 7 木屋场景4×4超分辨率对比图

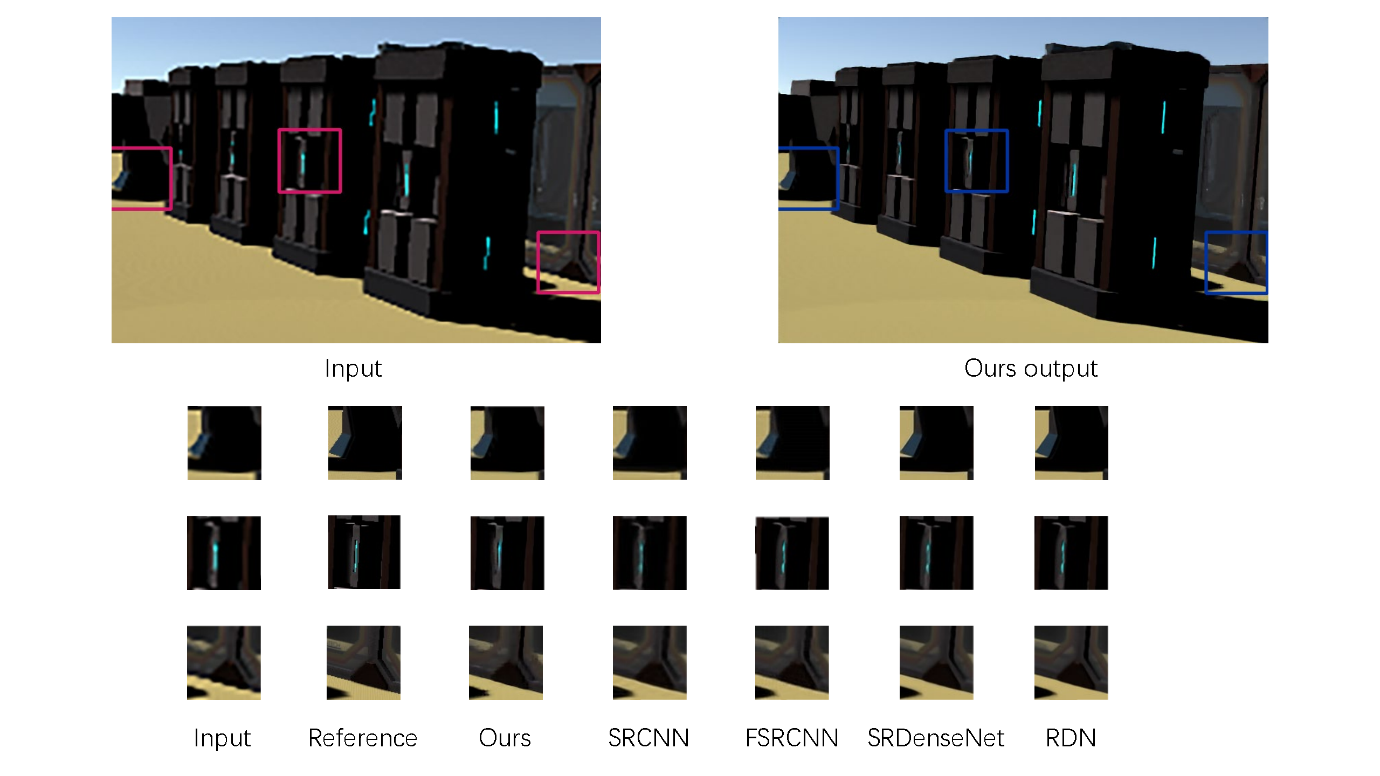


图 9 沙漠场景4×4超分辨率对比图

**4×4超分辨率渲染细节分析：**

1. 对于木屋场景：第一处是水井的辘轳，；第二处是屋顶水平木质横梁最低端的小凹槽；第三处是树边缘，可以看出本课题组的方法显著由于SRCNN和FSRCNN，在某些细节可能不如SRDenseNet和RDN，但这二者无法应用于实时渲染，本课题组会在后文通过比较这几者的渲染时间说明。
2. 对于游乐园场景：第一处为小孩的背影，可以清楚地看到不同方法的清晰程度差别；第二处为彩色屋顶，观察位置主要是屋顶的棱的清晰度；第三处为滑梯的通道表面，其颜色的鲜艳程度在每个方法中也不尽相同，与第一个场景类似，本课题组的方法在清晰度和颜色的还原程度达到一个较好的结果。
3. 对于沙漠场景：第一处为一块斜着的蓝色表面，蓝色表面的边界的清晰度可以反映方法细节的好坏；第二处为一条蓝色细缝，细缝较能反映超分辨率时抗锯齿的效果，可以看到本课题组的方法在抗锯齿方面达到了比较好的效果；第三处为底座上面黑色条纹，不同方法超分辨率得到的条纹宽度不同，本课题组的方法与原图更为接近。

下面选取木屋场景展示2×2超分辨时各个方法的结果，如图 10所示

图 10 木屋场景2×2超分辨率对比图

可以看出，在超分辨率为2×2时，各个方法的效果并无明显差别，此模型的优势主要体现在高于2×2超分辨率渲染上。

## 效率与画质的比较

本课题组将此次实现的实时超分辨率模型与上述提到的现有的各种超采样算法进行渲染时间和画质上的比价，画质的比较主要通过比较图片的PSNR和SSIM完成。实验测试内容是超分辨率渲染一张原分辨率（2×2使用的是木屋场景、4×4使用的是沙漠场景）为720×480的画面。具体数据请见表格 3、表格 4。

1. **超采样率为2×2**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | FSRCNN | SRCNN | SRDenseNet | RDN | Ours |
| PSNR (dB) | 31.34 | 30.91 | 32.72 | 33.93 | 32.2 |
| SSIM | 0.950 | 0.943 | 0.961 | 0.973 | 0.963 |
| Time (s) | 0.62 | 0.80 | 18.41 | 27.42 | 1.31 |

表格 3 超采样率2×2时渲染时间与画质

在2×2超采样时，本课题组的模型达到一个优秀的效果，在不太多增加耗时的同时，渲染画质优于基于SRCNN的两种方法，并略差与SRDenseNet和RDN两种重量级模型（耗时较长）。

1. **超采样率为4×4**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | FSRCNN | SRCNN | SRDenseNet | RDN | Ours |
| PSNR (dB) | 28.8 | 28.4 | 31.0 | 32.0 | 31.0 |
| SSIM | 0.830 | 0.852 | 0.918 | 0.928 | 0.923 |
| Time (s) | 0.35 | 0.86 | 3.22 | 7.00 | 1.61 |

表格 4 超采样率4×4时渲染时间与画质

可以看出基于SRCNN（FSRCNN、SRCNN）的方法进行图片超分辨率快于本课题组的方法，本课题组分析之后认为应该是SRCNN模型结构比本课题组的模型更加简单，所需的计算量更小，故而更快，但是从渲染的画质质量来看，本课题组的方法比这二者更优越；对于SRDDenseNet 和 RDN 模型而言，对一张图的渲染时间达到了4秒以上，虽然他们的渲染画质的比本课题组的方法好，但和它们提升画质的程度并不值得多花近4倍的时间，换而言之，这二者是无法应用于实时渲染中。

# 特色与创新/ Distinctive or Innovation Points

本课题组的工作的主要参考算法是Neural supersampling for real-time rendering[1] 一文，原文是基于使用前4帧恢复当前帧的思想来设计的神经网络结构，并且原文只对比了使用前1或2或3或4帧来超分辨率当前帧，本课题组的网络是采用前5帧来超分辨率当前帧，进一步拓展了原文的实验范围，并独立完整地设计和进行了比较详尽的实验。从实验结果分析，本课题组实现的算法有以下两个特色（优点）

1. 本课题组实现了一个恰当的模型，使其在渲染时间和画质上取得一个较为合理的平衡，即在保证较高质量渲染画质的情况下，尽可能的减少了渲染时间，这种平衡性是较为显著地优于现有的部分超分辨率算法，所以此模型可适用于对游戏画面的实时渲染。
2. 此模型优化了超分辨率过程中普遍存在的抗锯齿问题，并提出一种新的插值方案。其创新原理是先对低分辨率画面（包括各个特征，比如rgb，motion，depth等）进行零上采样，在通过对物理原理上更加具有连续性的运动矢量进行双线性插值，再用完成插值的运动矢量，通过不断的warp迭代，将前各个帧的特征最终结合到当前帧上，最终重构得到超分辨率的当前帧。

References:

1. Xiao L, Nouri S, Chapman M, et al. Neural supersampling for real-time rendering[J]. ACM Transactions on Graphics (TOG), 2020, 39(4): 142: 1-142: 12.
2. Kurt Akeley. 1993. Reality engine graphics. In Proceeding of Computer Graphics and Interactive Techniques (SIGGRAPH). 109—-116.
3. Alexander Reshetov. 2009. Morphological antialiasing. In Proceedings of High Performance Graphics (HPG). 109––116.
4. Jorge Jimenez, Jose I. Echevarria, Tiago Sousa, and Diego Gutierrez. 2012. SMAA: Enhanced subpixel morphological antialiasing. Comput. Graph. Forum 31, 2pt1 (2012), 355—-364.
5. Andrew Edelsten, Paula Jukarainen, and Anjul Patney. 2019. Truly next-gen: Adding deep learning to games and graphics. In NVIDIA Sponsored Sessions (Game Developers Conference).
6. Christoph Schied, Anton Kaplanyan, Chris Wyman, Anjul Patney, Chakravarty R. Alla Chaitanya, John Burgess, Shiqiu Liu, Carsten Dachsbacher, Aaron Lefohn, and Marco Salvi. 2017. Spatiotemporal variance-guided filtering: Real-time reconstruction for path-traced global illumination. In Proc. High Performance Graphics (HPG). Article 2, 2:1–2:12 pages.
7. Anjul Patney, Marco Salvi, Joohwan Kim, Anton Kaplanyan, Chris Wyman, Nir Benty, David Luebke, and Aaron Lefohn. 2016. Towards foveated rendering for gaze-tracked virtual reality. ACM Trans. Graph. 35, 6 (2016), 179:1–179:12.
8. Chakravarty R. Alla Chaitanya, Anton S. Kaplanyan, Christoph Schied, Marco Salvi, Aaron Lefohn, Derek Nowrouzezahrai, and Timo Aila. 2017. Interactive reconstruction of Monte Carlo image sequences using a recurrent denoising autoencoder. ACM Transactions on Graphics (Proceedings of SIGGRAPH) 36, 4 (2017), 98:1–98:12.
9. Anton S. Kaplanyan, Anton Sochenov, Thomas Leimkuehler, Mikhail Okunev, Todd Goodall, and Rufo Gizem. 2019. DeepFovea: Neural reconstruction for foveated rendering and video compression using learned statistics of natural videos. ACM Trans. Graph. (Proceedings of SIGGRAPH Asia) 38, 4 (2019), 212:1–212:13.
10. Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. 2015. Image superresolution using deep convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence 38, 2 (2015), 295–307.
11. Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. 2016. Accurate image super-resolution using very deep convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 1646–1654.
12. Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. 2017. Enhanced deep residual networks for single image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. 136–144.
13. Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. 2016. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 1874–1883.
14. Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. 2017. Deep Laplacian pyramid networks for fast and accurate super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 624–632.
15. Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. 2018b. Residual dense network for image super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2472–2481.
16. Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. 2017. Densely connected convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 4700–4708.
17. Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. 2018a. Image super-resolution using very deep residual channel attention networks. In Proceedings of the European Conference on Computer Vision. 286–301.
18. Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. 2017. Real-time video super-resolution with spatiotemporal networks and motion compensation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 4778–4787.
19. Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. 2019. EDVR: Video restoration with enhanced deformable convolutional networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops.
20. Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. 2018. Frame-recurrent video super-resolution. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 6626–6634.
21. Mengyu Chu, You Xie, Laura Leal-Taixé, and Nils Thuerey. 2018. Temporally coherent gans for video super-resolution (tecogan). arXiv preprint arXiv:1811.09393 (2018)
22. Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. 2004. Image quality assessment: from error visibility to structural similarity. IEEE Transactions on Image Processing 13, 4 (2004), 600–612
23. Justin Johnson, Alexandre Alahi, and Li Fei-Fei. 2016. Perceptual losses for real-time style transfer and super-resolution. In European Conference on Computer Vision. 694–711.
24. Dong, Chao, et al. "Learning a deep convolutional network for image super-resolution." European conference on computer vision. Springer, Cham, 2014.
25. Dong, Chao, Chen Change Loy, and Xiaoou Tang. "Accelerating the super-resolution convolutional neural network." European conference on computer vision. Springer, Cham, 2016.
26. Tong, Tong, et al. "Image super-resolution using dense skip connections." Proceedings of the IEEE international conference on computer vision. 2017.
27. Zhang, Yulun, et al. "Residual dense network for image super-resolution." Proceedings of the IEEE conference on computer vision and pattern recognition. 2018.

时间安排与分工统计表

|  |  |  |  |
| --- | --- | --- | --- |
| **组员信息**（含组长） | | | |
| 学生姓名 | 吴雨桐 | 学 号 | 519030910256 |
| 项目分工 | 代码复现与实验、部分实验报告编写 | | |
| 学生姓名 | 余北辰 | 学 号 | 519030910256 |
| 项目分工 | 生成数据集、现有超分辨率算法、部分实验报告编写、制作PPT | | |
| 学生姓名 |  | 学 号 |  |
| 项目分工 |  | | |
| **时间安排/**  **Schedule** | 10.28：确定选题  10.29~11.22：研读原论文以及相关论文  11.23~12.17：完整复现代码，并生成数据集  12.18~12.20：训练模型，并找现有超分辨率算法  12.20~12.22：做实验，并完成报告编写  12.23~12.25：制作ppt与演示视频 | | |