

Deep Joint Demosaicking and Denoising

吴荣源

March 12, 2020

文章设计了一个 CNN 实现端到端地联合去噪与去马赛克方法。

文章架构解析

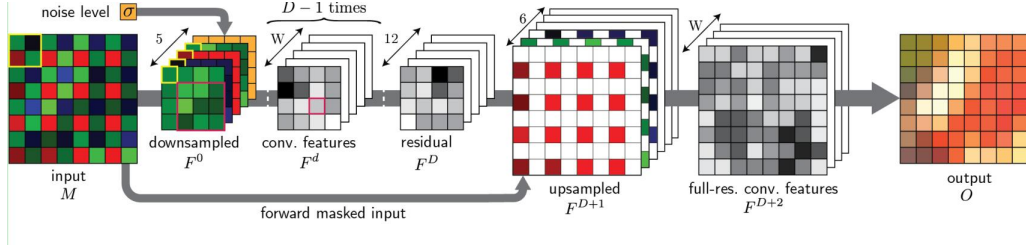


图 1: 联合去噪去马赛克网络架构

- 1) M 为马赛克输入图像, $G_r/B/R/G_b$ 为一个 2×2 的像素阵列, 每一类像素均作为一个通道被提取出来得到 F_0 。第五个输入通道为独立的加性高斯噪声, 方差为 $\sigma \in (\sigma_1, \sigma_2)$ 。在训练阶段, 对于每张输入马赛克图像, 噪声被随机取样添加作为第五个通道。
- 2) F_0 到 F_D 的训练过程中, 先经过了 $D-1$ 次卷积运算, 卷积核大小为 $K \times K \times W$; 再经过一次卷积, 卷积核大小为 $K \times K \times 12$, 用来调整最后的通道数为 12。
- 3) F_{D+1} 由两个方向组合而成。通道 1-3 是原始马赛克图 M 跨层连接而来, 通道 4-6 由 F_D 进行上采样得到。这有点借鉴残差网络的意思。
- 4) F_{D+1} 经过与步骤 2 中相同的卷积方式得到 F_{D+2} , 再经过卷积核大小为 $K \times K \times 1$ 的卷积得到输出图 O

作者对网络深度参数 D 从 5-20 均进行了实验, K 设置为 3, 卷积前后图像 height/width 不变

训练细节

- 1) 数据集由噪声 σ /RGB 图像 I /马赛克图像 M 组成, 损失函数为网络输出图像 O 与 RGB 图像 I 的 L_2 损失
- 2) 输入图像大小为 128×128 ; 初始权重和偏置设置为 0; 初始学习率设置为 10^{-4} ; 使用 ADAM 优化算法; batch size 设置为 64; 当验证集

上的损失经过 10 个 epoch 不再下降后，将学习率降低 10 倍，通常执行两次这种操作；使用 caffe 架构，在 NVIDIA Titan X 上训练 2-3 周。

数据库的选择

使用标准数据库进行训练时出现两个问题：

- 1) hard example 很少，大部分是平滑区域
- 2) 即使网络训练损失很低、通过网络得到的输出图像 O 的 PSNR 值很高，但是主观来看恢复图像还是会出现明显的 demosaicking artifacts

所以需要更改训练方式：

- 1) 在标准数据库上训练网络
- 2) 寻找存在 luminance artifacts/color moire 问题的图像块作为 hard example 对网络参数进行微调

下面详细介绍选取数据库的过程

- 1) 选取至少为 16M 的 sRGB 图像，为了避免相机去马赛克技术的影响，我们对原图进行两次双三次插值下采样作为 groundtruth。
- 2) 根据 Bayer 模式生成马赛克图像 M 并且随机添加噪声。
- 3) 通过旋转/左右镜像等方式增加数据集

‘挖掘难样本

文章称图像评价指标 HDR-VDP2 比起 PSNR/SSIM/S-CIELAB 更容易让人信服，故选择它来挖掘边缘很丰富的图像块。（这个评价指标只对亮度敏感，容易忽略 chrominance moire artifacts）

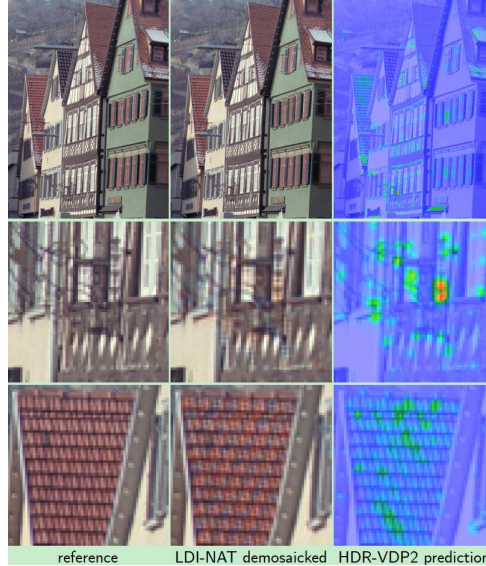


图 2: HDR-VDP2 用于去马赛克算法评价

操作方法：将马赛克图像经过预训练的网络得到输出图像 O ，用来与 groundtruth 比较输出 HDR-VDP2 预测热图（计算每个像素是 artifacts 的可能性）。然后使用高斯滤波器平滑（方差为 3），如果存在 30 个可能性大于 0.1 的区域我们将它当作难样本。从 1393107 张 15.2M 大小的图片中提取了总共有 2489180 个图像块当作难样本库（像素占比大约 0.03）

对于云纹的检查文章采用频域分析的方法，因为颜色插值失误会引入不可知的频率。200w 个图像块中只有 0.0005 的比例出现云纹，虽然比例很少，但是对于训练十分重要。

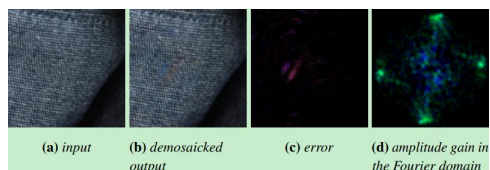


图 3: 云纹带来的频域增益

结果

在难样本库的训练过程中，D 被设置成 15，W 被设置成 64，K 被设置为 3。2590186 张 128x128 难样本图像当作训练集，4000 张图像验证集上的 loss 不下降后停止训练，测试集为另外 2000 张图像（一半为 vdp 样本，一半为云纹样本）

训练集和训练时间

我们将 imagenet 中的 1.3million 张图像当作训练集第一次训练我们的网络，虽然测试结果 PSNR 值很高，但是会产生很明显的视觉误差。文章认为这是样本不平衡造成的，平滑图像占据绝大多数。对第一次训练得到的网络使用难样本进行微调，或者重新开始训练结果差不多。