

工作总结与计划

一、工作总结（按时间划分）

日期	工作内容			完成情况及主要问题
	上午	下午	晚上	
星期一	处理杂事	选论文	读论文	简单对比后选择了 2020-TMM 的论文，并对论文要解决的问题有了了解
星期二	读论文	读论文	读论文	阅读了论文的主要方法和部分算法，基本了解论文方法的创新和构思
星期三	读论文	读论文	形教课	大体阅读完论文提出的方法及其作用，对自回归模型的使用不太清楚
星期四	处理杂事	读论文	休息	大致浏览了论文后续内容
星期五	读论文	处理杂事	复读论文	完成论文的阅读，对论文的实验流程和结论有了初步了解
星期六	复读论文	写周报	学习 pytorch	对论文进行了详细回顾，对论文要解决的问题及使用的方法有了更清楚的认识

二、下一步计划（按任务划分）

编号	工作内容	目标	相关配合
1	完成第四篇论文的阅读	11 月 27 日前完成	无
2	准备毕设任务书	12 月 05 日前完成	待定
3			
4			
5			
主要风险	无		

三、个人分析与总结

内容提要	
1	进度方面：本周能有效按照预期计划完成目标
2	课题方面：论文的解决问题相似，但方法差别较大
3	其他思考：在阅读论文之外要适当学习深度学习和 pytorch 等技术方面的内容
4	

四、论文总结

论文标题	Multi-Direction Dictionary Learning Based Depth Map Super-Resolution With Autoregressive Modeling
作者及单位	Jin Wang ¹ , Wei Xu ¹ , Jian-Feng Cai ² , Qing Zhu ¹ , Yunhui Shi ¹ , and Baocai Yin ³ 1. Beijing University of Technology 2. Hong Kong University of Science and Technology 3. Dalian University of Technology
论文出处	2020-TMM
创新点提炼	<p>本文的主要贡献可以总结如下：</p> <ol style="list-style-type: none">1) 提出了一种用于深度图表示的多方向字典学习方法（multi-direction dictionary learning method）。深度图中的边缘区域起着重要的作用，与自然图像相比，它们通常表现出独特的几何方向。现有的字典学习方法通常在随机选择的图像块上执行。但是，图像内容在图像之间可能会有很大差异。用各种 patch 训练的通用且过于完整的字典在表达输入信号方面既不是最佳方法，也不是有效方法。与通用字典相比，本文的多方向字典可以更精确地表示深度图中的几何方向；2) 自适应 AR 模型被用来以 pixel level 捕捉在深度图 patch 的小区域的局部模式。方向字典可以很好地表示面片级别中的几何方向。然而，与几何方向一致的局部模式在深度边缘 patch 中非常普遍。为了进一步在像素级别上表示这些局部模式，本文采用了 AR 模型。如果根据信号特征精心设计 AR 系数，AR 模型就能很好地拟合深度信号。3) 本文提出了一种将多方向字典和自适应 AR 模型相结合的复合深度图 SR 模型，并设计了一种有效的数值方案。该模型成功地从 patch level 和 pixel level 恢复了深度边缘。为了处理深度图和彩色图像之间的不一致，使用了逐像素乘积方法（pixel-wise product methods）。
个人想法	本文着力于深度图像中边缘区域的恢复，从几篇论文来看，边缘恢复在深度图像的超分辨率中发挥着至关重要的作用，同时，彩色图像能作为先验信息有效指导深度图像的超分辨率重建，但深度图与彩色图边缘不匹配是研究的难点。本文利用图像信号的分段统计稳态性对图像信号进行建模，从不同级别恢复了重建了深度图像。

论文方法及结论：

1、论文提出的问题

在深度图中，与自然图像相比**边缘区域**扮演着更重要的角色，并展现出独特的**几何方向**。但是，大多数现有的超分辨率方法都忽略了这一事实，并且它们无法正确处理深度边缘。

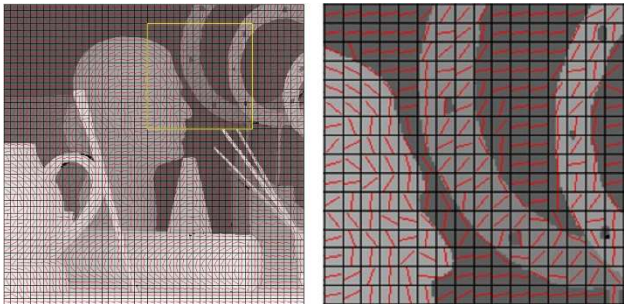


Fig. 1. Depthmap with estimated directions of images patches.

- 深度图的空间分辨率通常较低（例如 176×144 和 200×200 ），噪声严重，边缘附近有很多像素丢失。深度图的质量会严重影响许多现实世界中的多媒体应用中算法的性能。因此，如何提高 3D 相机捕获的原始深度图的质量是这些多媒体技术的基本问题和重要组成部分；
- 参考文献[6-19]的方法表明，彩色图像有助于改善深度图的质量——它使深度图中的边缘更明晰，平滑区域更平滑。但彩色图像中的边缘与深度图中的边缘并不完全匹配。尽管许多用来解决这一问题的方法可以在彩色图像的指导下恢复高质量的深度图，但它们通常专注于深度图的局部区域，而忽略了不同深度图的块间相关性；
- 基于示例的图像超分辨率（SR）方法得到了快速发展，其中高分辨率（HR）图像是通过从可用的低分辨率（LR）和 HR 图像补丁联合训练的字典中获得的。稀疏表示可以很好地表示 patch level 的深度图。但是，稀疏表示无法捕获与像素级别的几何方向一致的模式，尤其是对于深度边缘的小区域。
- 许多字典学习方法旨在学习一个完整的字典（over complete dictionary）来表示所有图像中包含的各种结构。但是，由于多种原因，通用的完整字典对于深度图表示既不是最佳方法也不是有效方法。
 - 由于深度图的大多数区域都是平滑的，边缘仅出现在一个很小的区域，因而平滑区域比边缘多得多。由于边缘比平滑区域包含更多的结构，因此无法对具有丰富结构的 patch 进行充分采样；
 - 由于图像结构在各个图像之间可能会发生显著变化，因此通用字典中针对一个图像的原子可能与某些其他图像无关。这些不相关的原子将降低表示精度和计算效率。
 - 为了表示所有图像中的大量结构，通用词典（universal dictionary）必须极其冗余。但是，高度冗余的字典上的稀疏表示形式可能会不稳定，并且容易产生视觉伪像。

2、解决的方法

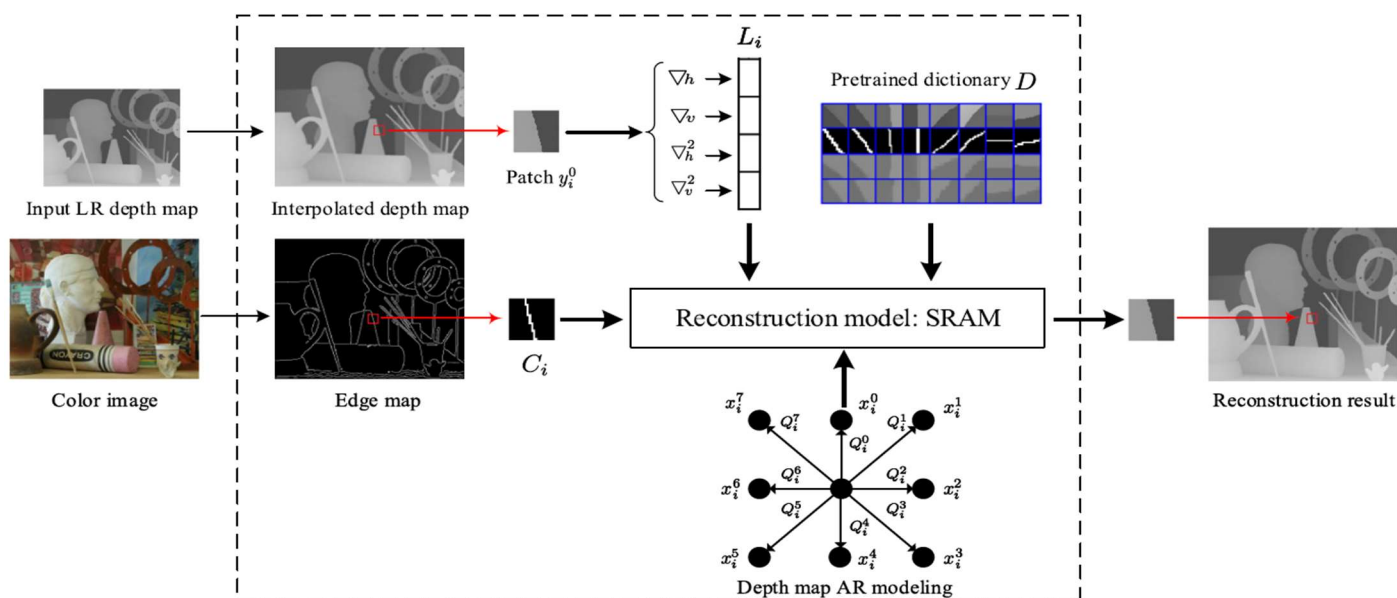


Fig.2. Depth map super-resolution framework. Given an input low resolution depth map and corresponding color image, we extract features from both of them respectively. With the pretrained multi-direction dictionary and adaptive ARmodel, we can infer a high resolution depth map via our reconstruction model (SRAM).

本文提出了一种基于多方向字典学习（multi-direction dictionary learning）的深度图 SR 方法。利用多方向词典和自回归（AR）模型来精确表示不同级别的深度边缘。

- 1) 在 patch level，具有几何方向的深度边缘块由预训练的多方向字典很好地表示；
- 2) 在 pixel level，自适应 AR 模型用于表示小区域中的局部相关模式。

为了更好地重建深度图，本文设计了一种复合深度图恢复模型，该模型结合了方向的稀疏表示和自适应 AR 模

型，以表示色 patch level 和 pixel level 的结构。

在本文的方法中，首先根据几何方向对深度图的 patch 进行分类，然后在每个类中训练获得一个稀疏字典。为了获得输入图像 patches 的稀疏表示，进一步将所有从不同几何方向训练的这些字典连接在一起，以形成新的字典。此外，为获得具有更清晰边界和更少错误边缘的 HR 深度图，引入彩色图像的纹理边缘信息。在构建阶段，采用自适应 AR 模型作为正则化项，这对于保持边缘清晰度非常有帮助。通过交替图像块及其稀疏系数进行最小化，可以从数值上解决所得的优化问题。

本文的主要贡献可以总结如下：

- 4) 提出了一种用于深度图表示的多方向字典学习方法（multi-direction dictionary learning method）。深度图中的边缘区域起着重要的作用，与自然图像相比，它们通常表现出独特的几何方向。现有的字典学习方法通常在随机选择的图像块上执行。但是，图像内容在图像之间可能会有很大差异。用各种 patch 训练的通用且过于完整的字典在表达输入信号方面既不是最佳方法，也不是有效方法。与通用字典相比，本文的多方向字典可以更精确地表示深度图中的几何方向；
- 5) 自适应 AR 模型被用来以 pixel level 捕捉在深度图 patch 的小区域的局部模式。方向字典可以很好地表示面片级别中的几何方向。然而，与几何方向一致的局部模式在深度边缘 patch 中非常普遍。为了进一步在像素级别上表示这些局部模式，本文采用了 AR 模型。如果根据信号特征精心设计 AR 系数，AR 模型就能很好地拟合深度信号。
- 6) 本文提出了一种将多方向字典和自适应 AR 模型相结合的复合深度图 SR 模型，并设计了一种有效的数值方案。该模型成功地从 patch level 和 pixel level 恢复了深度边缘。为了处理深度图和彩色图像之间的不一致，使用了逐像素乘积方法（pixel-wise product methods）。具体而言，在本文提出的方法中建立了 patch 之间的两种连接：（i）HR 和 LR patch 之间的连接；（ii）一个图像 patch 内像素之间的连接。此外，同一类别的 patch 具有相似性，因此补丁的分类反映了非局部自相似性。

Algorithm 1: Dictionary Learning on Classified Patches

Initialization:

Initialize \mathbf{Y}^0 using bicubic interpolation;
Set parameters $\lambda_l, \lambda_f, \lambda_h$ and λ ;

Output:

Dictionaries D_l, D_f and D_h .

Main:

```

1: for  $i = 1, 2, \dots, I$  do
2:   Compute the  $i$ -th LR patch feature as in (12),
     corresponding HR patch feature as in (13) and color
     patch feature as in (14);
3:   Estimate geometric directions of the  $i$ -th LR patches
     as in (11);
   end
4:   Form classes of patches sharing the same direction;
5:   foreach geometric direction  $\omega \in \{\omega_1, \dots, \omega_R\}$  do
6:     Solving (15) for  $D_\omega^l, D_\omega^f$  and  $D_\omega^h$  using K-SVD
     algorithm;
7:     Output  $D_\omega^l, D_\omega^f$  and  $D_\omega^h$ .
   end
8: Collect all the  $D_\omega^l, D_\omega^f$  and  $D_\omega^h (\omega \in \{\omega_1, \dots, \omega_R\})$ 
   together to form three new dictionaries  $D_l, D_f$  and  $D_h$ 
   respectively.
```

Algorithm 2: Depth Map Reconstruction via SRAM

Initialization:

Initialize \mathbf{Y}^0 using bicubic interpolation, $\mathbf{m}_i = \text{mean}(\mathbf{y}_i^0)$
Set parameters $\beta^{(0)} = \mathbf{0}, \mu > 0, \tau > 0, \lambda > 0,$
 $\lambda_1 > 0, \lambda_2 > 0, \lambda_3 > 0, \lambda_4 > 0$ and $\lambda_5 > 0$;

Output:

Recovered HR depth map.

Main:

```

1: for  $i = 1, 2, \dots, I$  do
2:   Compute the  $i$ -th LR patch feature as (12), and color
     patch feature as (14);
3:   Initialize  $\mathbf{x}_i$  by solving (8);
4:   for  $m = 1, 2, \dots, M$  do
5:     Solving (24) using OMP for  $\alpha_i$ ;
6:     for  $n = 1, 2, \dots, N$  do
7:       Solving (23) for  $\mathbf{x}_i^{(l+1)}$  by using the conjugate
         gradient algorithm;
8:        $\beta^{(l+1)} = \beta^{(l)} + \mu^{(l)}(\mathbf{x}_i^{(l+1)} - D_h \alpha_i - \mathbf{m}_i)$ ;
9:        $\mu^{(l+1)} = \tau \cdot \mu^{(l)}$ ;
     end
   end
10: Output the recovered depth map patch  $\mathbf{x}_i$ ;
   end
11: Recover the whole HR depth map by combining all the
     patches  $\mathbf{x}_i$  obtained in the previous step.
```

2.1 MULTI-DIRECTION DICTIONARY LEARNING FOR DEPTH MAP SR

1) Image Patch Direction Estimation

在本论文方法的第一阶段，image patches 根据他们的几何方向被分为不同的类。我们通过以下方法来对其方向进行估计：

用 x_j 表示深度图 X 中的第 j 个 patch。 x_j 的方向 ω_j 通过以下方式估算：

$$\min_{\omega \in \{\theta_1, \dots, \theta_Q\}} \|\hat{c}_{j,\omega} - W^T G_\omega x_j\|_2^2$$

其中， G_ω 是沿着候选方向重新排列像素的运算符， $\{\theta_1, \theta_2, \dots, \theta_R\}$ 形成 R 个候选方向的一维矢量， W 是前向正交一维 Haar 小波， $\hat{c}_{j,\omega}$ 是幅度前 25% 大的小波系数。

最佳几何方向在所有候选方向之间提供最佳稀疏性。

2) Feature Representation

对于输入的 LR，本文没有直接将原始深度图 patch 用作输入，而是在水平和垂直方向上都使用一阶和二阶导数作为高通滤波器来表示 LR 深度 patch 的特征：

$$l_i = \left[\frac{\partial}{\partial h} y_i^0, \frac{\partial}{\partial v} y_i^0, \frac{\partial^2}{\partial h^2} y_i^0, \frac{\partial^2}{\partial v^2} y_i^0 \right]$$

其中， y_i^0 是输入深度图 Y 的插值结果 Y_0 的 patch。高通滤波器通常用于低级统计学习任务中，以增强有意义的特征。通过该方法获得每个 patch 的特征向量。

给定用于字典训练的输入 HR 深度图 X ，我们可以将其特征 h_i 表示为：

$$h_i = x_i - \text{mean}(x_i)$$

其中， x_i 是 X 的一个 patch， $\text{mean}(x_i)$ 是 x_i 的平均值。由于不知道 x_i ，因而不知道 $\text{mean}(x_i)$ 。由于 LR y_i^0 和 x_i 具有相似的均值，因此可以使用 y_i^0 的深度均值，由 m_i 近似表示 $\text{mean}(x_i)$ ，即 $x_i = h_i + x_i$ 。

为了消除彩色图像中的某些与深度图中不一致的边缘，提取彩色图像 C 和插值后的深度图像 Y_0 的边缘，然后将像素乘积作为彩色图像块的特征

$$f_i = (\nabla c_i) \times (\nabla y_i^0)$$

其中， ∇ 是边缘提取操作符， c_i 是彩色图像 C 的一个 patch。

3) Dictionary Learning

在字典学习中，仅涉及具有足够结构的 patch。如果 patch 中元素的方差大于阈值 Γ ($\Gamma=3$)，则该 patch 用于字典训练；否则，将其丢弃。通过这种方式，可以排除平滑区域的 patch，并且仅将具有一定数量边缘结构的 patch 用于训练。

然后根据 1) 所述的方法将合格的 patch 按照其几何方向分为几类。令 R 为 patch 类别的数量，而 S_ω ， $\omega = 1, \dots, R$ 是类 ω 中的 patch 集。然后从 S_ω 中提取 2) 中描述的特征，用 L_ω ， H_ω 和 F_ω 表示从 S_ω 提取的特征矩阵。最终目标是分别从 L_ω ， H_ω 和 F_ω 学习三个字典 D_ω^l ， D_ω^h ， D_ω^f 。

参考文献[61][34]假设对齐对应结构时，不同的域共享相似的稀疏系数。本文利用低分辨率深度图，高分辨率深度图和相应的彩色图像之间的这种依赖性来重构和细化深度图。该假设是合理的，因为由于常见的场景结构，RGB 图像和深度图中的不连续性通常会重合。此外，通过 2) 中方法消除了彩色图像中的边缘与对应的深度图之间的不一致。因此，本文约束 LR，HR 和彩色图像 patch 的特征在它们各自的字典的表示下分别具有相同的稀疏系数。即

$$\min_{D_{\omega}^l, D_{\omega}^f, D_{\omega}^h, A} \lambda_l \|L_{\omega} - D_{\omega}^l A\|_F^2 + \lambda_f \|F_{\omega} - D_{\omega}^f A\|_F^2 + \lambda_h \|H_{\omega} - D_{\omega}^h \alpha_i\|_F^2 + \lambda \|A\|_1$$

其中, $A = [\alpha_1, \alpha_2, \dots, \alpha_N]$ 是 $\|A\|_1$ 促进稀疏性的系数矩阵, λ_l , λ_f , λ_h 和 λ 是平衡系数。其可以通过标准的字典学习算法 (例如 K-SVD) 解决。

2.2 SPARSE RECONSTRCUTION WITH ADAPTIVE AR MODELING(SRAM)

AR 模型的关键见解在于, 信号可以由信号本身重新生成。在 AR 模型中, 图像信号表示如下

$$X(i, j) \approx \sum_{(m, n) \in T_M} \alpha(m, n) X(i + m, j + n)$$

其中 $\alpha(m, n)$ 是 AR 参数, T_M 是用于回归操作的空间模板, 通常包含局部邻域窗口中的像素。因而需要调整参数 α 以适合图像像素的局部结构。根据 AR 模型, 在一个小的局部窗口中, 模型参数 $\alpha(m, n)$ 保持恒定或接近保持不变。因此, 图像内容 (例如边缘和表面纹理) 是由空间相关的连续像素的固定线性组合形成的。换句话说, AR 模型假定图像信号是分段统计平稳的。分段平稳性使得可以通过将局部窗口的样本自适应地拟合到 AR 模型来学习边缘和纹理中的局部像素结构。

对于深度图 Y 中的每个输入的低分辨率 patch y_i , 通过插值获得其初始值 y_i^0 。然后通过之前介绍的方法得到其特征 l_i 和对应的彩色图特征 f_i 。如果 y_i^0 的平均值为 m_i , 则可以通过以下最小化函数获得其 HR 输出 x_i :

$$\min_{x_i, \alpha_i} \lambda_1 \|l_i - D_l \alpha_i\|_2^2 + \lambda_2 \|f_i - D_f \alpha_i\|_2^2 + \lambda_3 \|y_i - P x_i\|_2^2 + \lambda_4 \|x_i - Q_i x_i\|_2^2 + \lambda_5 \|\alpha_i\|_1 \text{ s.t. } x_i = D_h \alpha_i + m_i$$

其中, P 是观察算子矩阵, Q 是预测矩阵, 并且 λ_1 , λ_2 , λ_3 , λ_4 和 λ_5 是平衡参数。

该最小化问题可以通过交替 x_i 及其稀疏系数 α_i 进行最小化来解决。对于固定的稀疏系数 α_i , 可以优化 x_i :

$$\hat{x}_i = \arg \min_{x_i} \lambda_3 \|y_i - P x_i\|_2^2 + \lambda_4 \|x_i - Q_i x_i\|_2^2 \text{ s.t. } x_i = D_h \alpha_i + m_i$$

采用增强拉格朗日乘数 (Augmented Lagrange Multiplier, ALM) 算法来求解该问题。其增强拉格朗日函数为

$$L(x, \beta, \mu) = \lambda_3 \|y_i - P x_i\|_2^2 + \lambda_4 \|x_i - Q_i x_i\|_2^2 + \langle \beta, x_i - D_h \alpha_i - m_i \rangle + \mu \|x_i - D_h \alpha_i - m_i\|_2^2$$

其中, 其中 $\langle \cdot, \cdot \rangle$ 是内积, β 是拉格朗日乘数, μ 是正标量。通过 ALM 方法解决它将生成以下迭代:

$$x_i^{(l+1)} = \arg \min_{x_i} L(x_i, \beta^{(l)}, \mu^{(l)})$$

$$\beta^{(l+1)} = \beta^{(l)} + \mu^{(l)} (x_i^{(l+1)} - D_h \alpha_i - m_i)$$

$$\mu^{(l+1)} = \tau \cdot \mu^{(l)}$$

其中 $\tau > 1$ 为常数。对于固定的 $\beta^{(l)}$ 和 $\mu^{(l)}$, 通过取 $\partial L(x_i, \beta^{(l)}, \mu^{(l)}) / \partial x_i = 0$ 来求解第一个式子, 得到:

$$x_i = [\lambda_3 P^T P + \lambda_4 (I - Q_i)^T (I - Q_i) + \mu I]^{-1} \times \left(\lambda_3 P^T y_i - \frac{\beta}{2} + \mu (D_h \alpha_i + m_i) \right)$$

上式右侧要反转的矩阵是一个正方形矩阵, 为提高效率, 使用共轭梯度 (conjugate gradient, CG) 算法来

计算 x_i 。所有 x , β , μ 都更新直到收敛。

对于固定的 x_i , 稀疏系数 α_i 可以通过

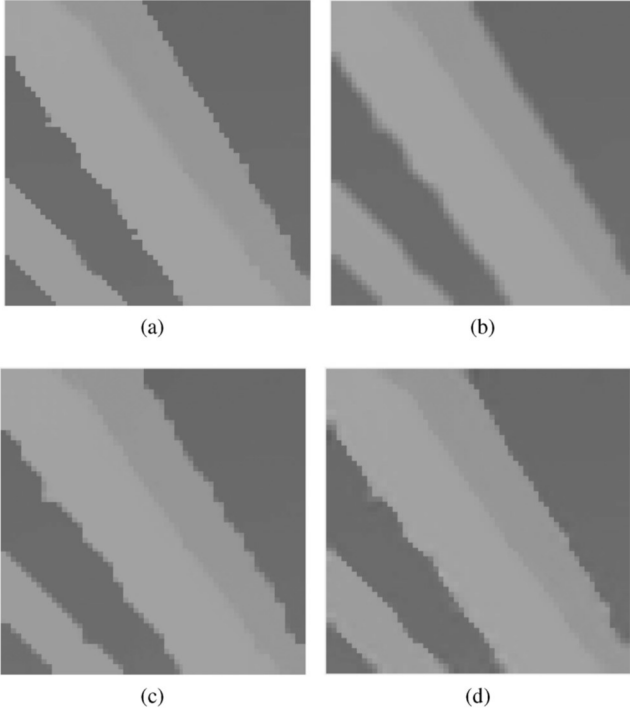
$$\hat{\alpha}_i = \arg \min_{\alpha_i} \lambda_1 \|l_i - D_l \alpha_i\|_2^2 + \lambda_2 \|f_i - D_f \alpha_i\|_2^2 + \lambda_5 \|\alpha_i\|_1 \text{ s.t. } x_i = D_h \alpha_i + m_i$$

上式可以近似为:

$$\hat{\alpha}_i = \arg \min_{\alpha_i} \lambda_1 \|l_i - D_l \alpha_i\|_2^2 + \lambda_2 \|f_i - D_f \alpha_i\|_2^2 + \lambda \|x_i - D_h \alpha_i - m_i\|_2^2 + \lambda_5 \|\alpha_i\|_1$$

上式可以通过正交匹配追踪 (Orthogonal Matching Pursuit, OMP) 算法解决。

3、实验结果结论



A test result of SRAM is shown in Fig. 3(d), and it is compared with the results by the sparse representation model (Fig. 3(b)) and the AR model (Fig. 3(c)). We can see that the sparse representation model (Fig. 3(b)) achieves lower Root Mean Squared Error (RMSE) than the AR model, and the AR model obtain a depth map with sharper edges (Fig. 3(c)) than the sparse representation model (Fig. 3(b)). Our proposed SRAM (combination of the sparse representation and the AR model) gives a result that combines advantages of both methods. SRAM achieves a reconstruction (Fig. 3(d)) closest to the ground truth (Fig. 3(a)), with a lowest RMSE (RMSE = 1.99)

Fig.3. Reconstructed test image (upsampling factor $s = 2$).
(a) Ground truth. (b) Sparse representation model (RMSE=3.89).
(c) AR model (RMSE=4.86). (d) Proposed model (RMSE = 1.99).

Experiments on Middlebury Dataset: Table I and Table II show the quantitative comparison of results of upsampling from noise free depthmaps in terms of MAD and RMSE. The best performance are marked bold. Nevertheless, the MAD scores of our proposed method (rank around 1st to 3rd) are highly competitive with other state-of-the-art methods. In terms of RMSE, our proposed method achieves the lowest RMSE for all the testing depth maps with different scaling factors. The Edge method [22] provides slightly better results at low upsampling factors (especially on Reindeer, Laundry and Dolls). Dictionary learning method JESR [31] has similar performance with SCSR [27] under upsampling factor 2. However, when the upsampling factor increases, JESR [31] performs better than SCSR [27]. This is mainly because of the assistance of color image information. In visual comparison, our proposed method achieves more detailed structures and fewer false edges than others.

A Gaussian noise with a variance of 5 is added to the original images in the dataset, and then we downsample them at a subsampling rate of 4. Quantitative comparison results are given in Table III, where CLMF stands for the zero-order polynomial model. Our approaches outperform other compared state-of-the-art algorithms with the lowest MAD. The state-of-the-art methods JESR [31], TGV [64], AR [21] and our two methods perform well in suppressing noise, while our methods also perform very well in suppressing the ghost visual artifacts. The structures of the scene in boundary regions are preserved better with fine details in our methods than the others.

Experiments on Real Datasets: our proposed method in this paper achieves the best result in terms of MAD in mm. Moreover, our proposed model in this paper gives fewer false edges and little black holes in the restoration results. (ToF Depth Maps) Our method achieves better results than AR [21] and [46] with clearer boundary at depth discontinuities and smoother non-edge regions. (Kinect DepthMaps)

TABLE I
MAD COMPARISON ON MIDDLEBURY DATASET WITH DIFFERENT METHODS AT FOUR UPSAMPLING RATES

	Art				Book				Moebius				Reindeer				Laundry				Dolls			
	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
GIF[12] 2013	0.63	1.01	1.70	3.46	0.22	0.35	0.58	1.14	0.23	0.37	0.59	1.16	0.42	0.53	0.88	1.80	0.38	0.52	0.95	1.90	0.28	0.35	0.56	1.13
SRF[67] 2015	0.46	0.97	1.83	3.44	0.15	0.32	0.59	1.12	0.14	0.32	0.51	1.10	0.30	0.55	1.04	1.85	0.23	0.54	1.06	1.99	0.20	0.35	0.64	1.19
Edge[22] 2011	0.41	0.65	1.03	2.11	0.17	0.30	0.56	1.03	0.18	0.29	0.51	1.10	0.20	0.37	0.63	1.28	0.17	0.32	0.54	1.14	0.16	0.31	0.56	1.05
SCSR[27] 2010	0.46	0.96	1.81	3.45	0.15	0.31	0.59	1.13	0.16	0.31	0.50	1.11	0.31	0.54	0.99	1.82	0.23	0.55	1.05	1.97	0.20	0.35	0.61	1.15
CLMF[68] 2012	0.44	0.76	1.44	2.87	0.14	0.28	0.51	1.02	0.15	0.29	0.51	1.09	0.32	0.51	0.84	1.55	0.30	0.50	0.80	1.67	0.23	0.34	0.60	1.01
JESR[31] 2012	0.45	0.76	1.51	2.98	0.15	0.27	0.48	0.90	0.16	0.30	0.44	1.01	0.31	0.47	0.69	1.42	0.23	0.50	0.96	1.47	0.20	0.32	0.51	0.92
JGF[11] 2013	0.29	0.47	0.78	1.54	0.15	0.24	0.43	0.81	0.15	0.25	0.46	0.80	0.23	0.38	0.64	1.09	0.21	0.36	0.64	1.20	0.19	0.33	0.59	1.06
A+[65] 2014	0.31	0.63	0.90	2.41	0.15	0.26	0.41	0.94	0.19	0.34	0.47	0.87	0.33	0.51	0.78	2.06	0.33	0.46	0.87	1.98	0.26	0.33	0.57	1.21
SRCNN[37] 2016	0.22	0.53	0.77	2.13	0.09	0.22	0.40	0.79	0.10	0.22	0.42	0.89	0.32	0.47	0.68	1.77	0.24	0.50	0.96	1.54	0.23	0.33	0.57	1.09
TGV[64] 2013	0.45	0.65	1.17	2.30	0.18	0.27	0.42	0.82	0.18	0.29	0.49	0.90	0.32	0.49	1.03	3.05	0.31	0.55	1.22	3.37	0.21	0.33	0.70	2.20
AR[21] 2014	0.18	0.49	0.64	2.01	0.12	0.22	0.37	0.77	0.10	0.20	0.40	0.79	0.22	0.40	0.58	1.00	0.20	0.34	0.53	1.12	0.21	0.34	0.50	0.82
[36] 2019	0.43	0.43	1.10	3.24	0.29	0.24	0.49	1.14	0.34	0.27	0.47	1.11	0.34	0.30	0.67	1.70	0.40	0.53	0.67	1.76	0.42	0.32	0.48	1.01
[35] 2018	0.61	1.61	2.43	4.79	0.25	1.06	0.95	1.89	0.22	1.01	0.97	1.95	0.35	1.21	1.46	2.96	0.33	1.18	1.36	2.68	0.24	1.02	1.02	1.98
[28] 2017	0.69	1.07	2.10	5.49	0.33	0.50	0.92	2.17	0.34	0.51	0.96	2.39	0.46	0.72	1.30	3.32	0.42	0.61	1.31	3.00	0.41	0.56	1.02	2.23
[23] 2016	0.72	1.07	1.74	3.59	0.41	0.53	0.76	1.47	0.39	0.53	0.80	1.56	0.46	0.62	0.90	2.18	0.40	0.53	0.83	2.02	0.53	0.69	0.94	1.60
[66] 2019	3.06	2.18	2.57	2.71	2.75	4.61	4.99	5.19	0.47	0.25	0.18	0.97	0.21	0.31	0.65	0.82	3.65	4.13	4.35	3.87	0.21	0.36	0.48	0.75
OUR[46] 2017	0.25	0.55	0.63	2.01	0.12	0.24	0.38	0.76	0.14	0.19	0.40	0.77	0.23	0.41	0.55	1.01	0.21	0.35	0.54	1.15	0.22	0.27	0.51	0.83
OUR2	0.17	0.46	0.62	1.87	0.14	0.24	0.37	0.73	0.13	0.19	0.37	0.74	0.22	0.41	0.55	0.95	0.19	0.32	0.53	1.11	0.20	0.29	0.51	0.79

TABLE II
RMSE COMPARISON ON MIDDLEBURY DATASET WITH DIFFERENT METHODS AT FOUR UPSAMPLING RATES

	Art				Book				Moebius				Reindeer				Laundry				Dolls			
	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
GIF[12] 2013	3.15	4.11	5.73	8.53	1.41	2.03	2.58	3.67	1.15	1.65	2.58	4.12	2.19	2.98	4.44	6.58	1.88	2.60	4.02	5.89	1.18	1.67	2.10	3.24
SRF[67] 2015	2.65	3.89	5.51	8.24	1.06	1.62	2.38	3.41	0.90	1.37	2.06	2.99	1.95	2.84	4.10	5.97	1.61	2.40	3.50	5.24	1.14	1.39	1.98	2.79
Edge[22] 2011	2.58	3.24	4.30	6.03	1.21	1.52	1.93	2.60	0.86	1.27	1.99	2.68	1.96	2.89	3.58	3.99	1.62	2.39	3.22	4.29	1.12	1.32	1.51	2.20
SCSR[27] 2010	2.63	3.86	5.46	8.14	1.05	1.60	2.33	3.31	0.87	1.33	1.96	2.84	1.95	2.82	4.00	5.85	1.61	2.40	3.44	5.04	1.14	1.34	1.88	2.65
CLMF[68] 2012	2.59	3.44	4.50	6.93	1.23	1.47	1.93	2.73	0.92	1.33	1.76	2.43	1.96	2.72	3.35	4.80	1.68	2.49	2.85	4.18	1.01	1.30	1.73	2.25
JESR[31] 2012	2.63	3.66	5.13	7.05	1.05	1.59	1.83	2.91	0.87	1.21	1.59	2.24	1.95	2.69	3.55	4.88	1.61	2.34	2.84	4.44	1.13	1.32	1.67	2.25
JGF[11] 2013	3.08	3.94	5.25	7.13	1.32	1.82	2.38	3.49	1.14	1.59	2.34	3.47	2.17	2.78	3.50	4.46	1.87	2.59	3.68	5.24	1.13	1.50	1.98	2.71
TGV[64] 2013	2.60	3.34	4.10	6.43	1.20	1.47	1.82	2.63	0.82	1.22	1.64	2.41	1.80	2.71	3.15	4.60	1.61	2.39	2.64	4.17	1.01	1.31	1.61	2.22
A+[65] 2014	2.64	3.93	5.45	8.02	1.23	1.53	1.89	2.77	0.87	1.28	1.99	2.75	2.09	2.93	4.01	5.69	1.81	2.62	3.88	4.21	1.15	1.34	1.87	2.77
SRCNN[37] 2016	2.63	3.53	5.34	7.68	1.20	1.47	1.84	2.84	0.86	1.20	1.87	2.67	2.07	2.78	3.54	4.86	1.67	2.18	2.78	4.49	1.15	1.33	1.66	2.64
AR[21] 2014	1.80	2.80	3.65	5.90	1.33	1.54	1.82	2.92	0.86	1.01	1.46	2.53	1.81	2.45	2.96	4.11	1.70	2.22	2.69	4.57	1.18	1.38	1.66	2.34
[36] 2019	1.52	2.04	3.54	6.78	0.77	0.80	1.64	2.89	0.92	0.87	1.41	2.47	1.13	1.39	2.80	4.80	1.12	1.34	2.47	4.28	1.03	0.97	1.32	2.13
[35] 2018	4.21	6.27	9.57	13.7	1.86	3.14	4.29	6.14	1.46	2.60	3.79	5.78	3.14	4.99	7.38	10.4	2.40	4.14	6.29	9.01	1.45	2.38	3.27	4.81
[28] 2017	3.06	4.34	6.06	9.94	1.29	1.94	2.72	4.46	1.13	1.78	2.62	4.52	2.27	3.37	4.49	6.59	1.91	2.72	3.89	6.32	1.17	1.65	2.28	3.69
[23] 2016	2.37	3.50	4.76	8.49	1.13	1.52	2.18	3.48	0.97	1.43	2.22	3.80	1.80	2.56	3.21	5.84	1.46	2.07	2.90	5.26	1.11	1.48	1.94	2.93
[66] 2019	3.68	2.79	3.69	4.76	3.17	5.28	5.80	6.27	0.77	0.66	0.82	1.64	0.50	0.93	1.63	2.48	3.79	4.35	4.67	4.36	0.49	0.81	1.11	1.53
OUR[46] 2017	1.73	2.61	3.43	5.29	1.14	1.52	1.77	2.81	0.77	0.92	1.22	2.21	1.69	2.23	2.66	3.74	1.56	2.11	2.38	4.20	1.08	1.29	1.47	2.11
OUR2	1.67	2.57	3.20	4.87	1.02	1.33	1.46	2.51	0.72	0.85	1.10	1.98	1.67	2.07	2.47	3.44	1.43	2.00	2.11	4.07	0.97	1.07	1.19	1.90

TABLE III
MAD COMPARISON ON NOISY SIMULATED TOF DATA AT FOUR UPSAMPLING RATES

	Art				Book				Moebius				Reindeer				Laundry				Dolls			
	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
GIF[12] 2013	1.49	1.97	3.00	4.91	0.80	1.22	1.95	3.04	1.18	1.90	2.77	3.55	1.29	1.99	2.99	4.14	1.28	2.05	3.04	4.10	1.19	1.94	2.80	3.50
SRF[67] 2015	1.21	2.37	3.59	5.01	0.83	1.57	1.88	2.77	1.34	1.60	2.33	3.35	1.23	1.76	2.46	3.90	1.09	1.82	2.76	3.71	1.22	1.63	2.26	3.15
Edge[22] 2011	1.69	2.40	3.60	5.75	1.12	1.44	1.81	2.59	1.13	1.45	1.95	2.91	1.20	1.60	2.40	3.97	1.28	1.63	2.20	3.34	1.14	1.54	2.07	3.02
SCSR[27] 2010	1.22	1.70	3.02	4.88	1.07	1.55	2.04	3.09	1.16	1.54	2.23	3.20	1.23	1.67	2.35	3.90	0.99	1.65	2.68	3.51	1.26	1.61	2.22	3.10
CLMF[68] 2012	1.19	1.77	2.95	4.91	0.90	1.48	2.38	3.36	0.87	1.44	2.32	3.30	0.96	1.56	2.54	3.85	0.94	1.55	2.50	3.81	0.96	1.54	2.37	3.25
JESR[31] 2012	1.22	1.64	2.71	4.35	1.08	1.49	1.85	2.86	1.16	1.43	2.00	2.48	1.22	1.61	2.13	3.51	0.99	1.53	2.38	3.02	1.24	1.50	1.98	3.09
JGF[11] 2013	2.36	2.74	3.64	5.46	2.12	2.25	2.49	3.25	2.09	2.24	2.56	3.28	2.18	2.40	2.89	3.94	2.16	2.37	2.85	3.90	2.09	2.22	2.49	3.25
TGV[64] 2013	0.82	1.26	2.76	6.87	0.50	0.74	1.49	2.74	0.56	0.89	1.72	3.99	0.59	0.84	1.75	4.40	0.61	1.59	1.89	4.16	0.66	1.63	1.75	3.71
AR[21] 2014	0.76	1.01	1.70	3.05	0.47	0.70	1.15	1.81	0.46	0.72	1.15	1.92	0.48	0.80	1.29	2.02	0.51	0.85	1.30	2.24	0.59	0.91	1.32	2.08
[36] 2019	3.93	4.03	4.46	5.83	3.85	3.92	4.09	4.40	3.87	3.92	4.07	4.34	3.93	3.94	4.23	4.84	3.89	3.96	4.20	4.76	3.87	3.96	4.05	4.25
[35] 2018	3.05	1.64	2.63	4.79	2.76	1.07	1.17	1.89	2.72	1.03	1.19	1.97	2.84	1.23	1.64	2.98	2.80	1.20	1.58	2.68	2.74	1.04	1.21	1.98
[28] 2017	1.09	1.35	2.16	5.47	0.76	0.68	0.98	2.15	0.79	0.73	1.02	2.35	1.03	0.93	1.39	3.31	0.80	0.84	1.37	2.97	0.87	0.74	1.07	2.21
[23] 2016	0.70	1.01	1.61	3.93	0.54	0.75	1.07	1.99	0.54	0.76	1.16	2.00	0.57	0.80	1.18	2.55	0.55	0.77	1.18	2.44	0.65	0.87	1.22	1.88
[66] 2019	4.95	2.68	2.47	2.54	4.16	3.82	4.47	4.93	3.84	1.38	0.62	0.85	4.06	1.68	0.82	0.63	4.09	4.62	4.39	3.57	7.69	6.59	6.27	7.58
OUR[46] 2017	0.71	0.84	1.64	2.88	0.43	0.72	1.02	1.57	0.51	0.70	0.99	1.78	0.46	0.71	1.20	1.93	0.50	0.77	1.23	2.01	0.48	0.77	1.19	1.95
OUR2	0.64	0.71	1.32	2.43	0.39	0.66	0.96	1.46	0.46	0.64	0.83	1.57	0.40	0.65	1.09	1.87	0.47	0.71	1.20	1.86	0.44	0.69	1.11	1.64