Depth Map Super-Resolution by Deep Multi-Scale Guidance

Tak-Wai Hui[1], Chen Change Loy[1,2(✉)], and Xiaoou Tang[1,2]

[1] Department of Information Engineering,
The Chinese University of Hong Kong, Sha Tin, Hong Kong
{twhui,ccloy,xtang}@ie.cuhk.edu.hk
[2] Shenzhen Institutes of Advanced Technology,
Chinese Academy of Sciences, Shenzhen, China

Abstract. Depth boundaries often lose sharpness when upsampling from low-resolution (LR) depth maps especially at large upscaling factors. We present a new method to address the problem of depth map super resolution in which a high-resolution (HR) depth map is inferred from a LR depth map and an additional HR intensity image of the same scene. We propose a Multi-Scale Guided convolutional network (MSG-Net) for depth map super resolution. MSG-Net complements LR depth features with HR intensity features using a multi-scale fusion strategy. Such a multi-scale guidance allows the network to better adapt for upsampling of both fine- and large-scale structures. Specifically, the rich hierarchical HR intensity features at different levels progressively resolve ambiguity in depth map upsampling. Moreover, we employ a high-frequency domain training method to not only reduce training time but also facilitate the fusion of depth and intensity features. With the multi-scale guidance, MSG-Net achieves state-of-art performance for depth map upsampling.

1 Introduction

The use of depth information of a scene is essential in many applications such as autonomous navigation, 3D reconstruction, human-computer interaction and virtual reality. The introduction of low-cost depth camera facilitates the use of depth information in our daily life. However, the resolution of depth maps which is provided in a low-cost depth camera is generally very limited. To facilitate the use of depth data, we often need to address an upsampling problem in which the corresponding high-resolution (HR) depth map is recovered from a given low-resolution (LR) depth map.

Depth map super-resolution is a non-trivial task. Specifically, fine structures in HR image are either lost or severely distorted (depending on the scale factor used) in LR image because they cannot be fully represented by the limited spatial resolution. A brute-force upsampling of LR image simply causes those structures which are supposed to have sharp boundaries become blurred in the upsampled image. Ambiguity in super-resolving the severely distorted fine structures often

B. Leibe et al. (Eds.): ECCV 2016, Part III, LNCS 9907, pp. 353–369, 2016.
DOI: 10.1007/978-3-319-46487-9_22

(a) (b) (c) (d) (e)

Fig. 1. Ambiguity in upsampling depth map. (a) Color image. (b) Ground truth. (c) (Enlarged) LR depth map downsampled by a factor of 8. Results for upsampling: (d) SRCNN [11], (e) Our solution without ambiguity problem.

(a) (b) (c) (d) (e)

Fig. 2. Over-texture transfer in depth map refinement and upsampling using intensity guidance. (a) Color image. (b) Ground truth. (c) Refinement of (b) using (a) by Guided Filtering [8] ($r = 4, \epsilon = 0.01^2$). Results of using (a) to guide the 2× upsampling of (b): (d) Ferstl *et al.* [4], (e) Our solution.

exists, especially for the case of single-image upsampling. Figure 1(c–d) demonstrates the upsampling ambiguity problem.

To address the aforementioned problem, a corresponding intensity image[1] is often used to guide the upsampling process [1–7] or enhance the low-quality depth maps [8–10]. This is due to the fact that a correspondence between an intensity edge and a depth edge can be most likely established. Since the intensity image is at a higher resolution, its intensity discontinuities can be used to locate the associated depth discontinuities in a higher resolution. Although there could be an exception that an intensity edge does not correspond to a depth edge or vice versa, this correspondence assumption has been used widely in the literature.

One would encounter issues too in exploiting the intensity guidance. Specifically, suppose we have a perfectly registered pair of depth map D and intensity image Y possessing the *same* resolution. It is not straight forward to use Y to guide the refinement of D or the upsampling of LR D. The variation of depth structures in D may not be consistent with that of the intensity structures in Y as they are different in nature. Using image-guided filtering, features in intensity images are often over-transferred to the depth image at the boundaries between textured and homogeneous regions. Figure 2(c–d) illustrates two examples for the over-texture transferring problem. Our proposed method that complements D with only consistent structures from Y can avoid this problem (Fig. 2(e)).

[1] Intensity image represents either a color or grayscale image. We only study grayscale image in this paper.

In this paper, we present a novel end-to-end upsampling network, a Multi-Scale Guided convolutional network (MSG-Net), which learns HR features in the intensity branch and complements the LR depth structures in the depth branch to overcome the aforementioned problems. MSG-Net is appealing in that it allows the network to learn rich hierarchical features at different levels. This in turn makes the network to better adapt for upsampling of both fine- and large-scale structures. At each level, the upsampling of LR depth features is closely guided by the associated HR intensity features possessing the same resolution. The integrated multi-scale guidance progressively resolves ambiguity in depth map upsampling. We further present a high-frequency training approach to reduce training time and facilitate the fusion of depth and intensity features. Note that unlike existing super-resolution networks [11,12] that require pre-upsampling of input image by a conventional method such as bicubic interpolation *outside* the network. Our approach learns upsampling kernels *inside* a network to fully explore the upsampling ability of a CNN. We show that such a multi-scale upsampling method uses a more effective way to upscale LR images, while capable of exploiting the guidance from HR intensity features seamlessly.

Contributions: (1) We propose a new framework to address the problem of depth map upsampling by complementing a LR depth map with the corresponding HR intensity image using a convolutional neural network in a multi-scale guidance architecture (MSG-Net). To the best of our knowledge, no prior studies have proposed this idea for CNN before. (2) With the introduction of multi-scale upsampling architecture, our compact single-image upsampling network (MS-Net) in which no guidance from HR intensity image is present already outperforms most of the state-of-the-art methods requiring guidance from HR intensity image. (3) We discuss detailed steps to enable both MSG-Net and MS-Net to perform image-wise upsampling and end-to-end training.

2 Related Work

There is a variety of methods to perform image super resolution in the literature. Here, we categorize them into four groups:

Local methods are based on filtering. Yang *et al.* used the joint bilateral filter [1] to weight the degree of smoothing in each depth patch by considering the color similarity between the center pixel and its neighborhood [13]. Liu *et al.* designed the upsampling weights using geodesic distances [14]. With the use of image segmentation, Lu *et al.* developed a smoothing method to reconstruct depth structures within each segment [6].

Global methods formulate depth upsampling as an optimization problem where a large cost is given to a pixel in depth map if neighboring depth pixels have similar color in the associated intensity image but different depth values. Diebel *et al.* proposed Markov Random Field (MRF) formulation, which consists of a data term from LR depth map and a smoothness term from the corresponding HR intensity image for depth upsampling [15]. Park *et al.* utilized nonlocal

means filtering in which intensity features are acted as weights in depth regularization [2]. Ferstl *et al.* used an anisotropic diffusion tensor to regularize depth upsampling [4]. Yang *et al.* developed an adaptive color-guided auto regression model for depth recovery [5]. Aodha *et al.* especially focused on single-image upsampling as MRF labeling problem [16].

Dictionary methods exploit the relationship between a paired LR and HR depth patches through sparse coding. Yang *et al.* sought the coefficients of this representation to generate HR output [17]. Timofte *et al.* improved sparse-coding method by introducing the anchored neighborhood regression [18]. Ferstl *et al.* proposed to learn a dictionary of edge priors for an anisotropic guidance [19]. Li *et al.* proposed a joint examples-based upsampling method [20]. Kwon *et al.* formulated an upscaling problem which consists of scale-dependent dictionaries and TV regularization [7].

CNN-based methods are in distinction to dictionary-based approaches in that CNN do not explicitly learn dictionaries. With the motivation from convolutional dictionaries [21], Osendorfer *et al.* presented a convolutional sparse coding method for super-resolving images [22]. Wang *et al.* developed a cascade of sparse coding based networks (CSCN) [12] that are constructed by using modules from the network for the learned iterative shrinkage and thresholding algorithm (LISTA) [21]. However, their decoder uses sparse code to infer a HR patch separately. All the recovered patches are required to put back to the corresponding positions in HR image. Dong *et al.* proposed an end-to-end super-resolution convolutional neural network (SRCNN) to achieve image restoration [11].

Comparing to the above methods, our CNNs exhibit several advantages. We do not explicitly formulate an optimization problem as the global methods [2,4,5,15] or design a fixed filter as the local methods [6,13,14] because CNN can be trained to address the upsampling problem. In contrast to the dictionary methods [7,19], our networks are self-regularized. No extra regularization on the upsampled image is necessary outside the network. In distinction to other single-image super resolution CNNs [11,12,22], our networks do not use a single $fixed$ (non-trainable) upsampling operator. More importantly, our MSG-Net is specifically designed for image-guided depth upsampling. Rich hierarchical features in the HR intensity image are learned to guide the upsampling of the LR depth map progressively in multiple levels towards the desired HR depth map. The multi-scale fusion architecture in turn enables MSG-Net to achieve high-quality upsampling performance especially at large upscaling factors.

Our work is related to the multi-scale CNNs for semantic segmentation (FCN) [23], inferring images of chairs [24], optical flow generation (FlowNet) [25] and holistically-nested edge detection (HED) [26]. Our network architecture differs from theirs significantly. An upsampling network is used in [24]. A downsampling network is used in HED. A downsampling sub-network followed by an upsampling sub-network is used in FlowNet and FCN. We use an upsampling (depth) branch in parallel with a downsampling (intensity) branch. This network architecture has not been studied yet. In common to [23–25], we use multiple

backwards convolutions for upsampling. But we do not use feed-fowarding and unpooling. All the above networks do not use deep supervision except HED.

3 Intensity-Guided Depth Map Upsampling

Suppose we have a LR depth map D_l which is down-sampled from its HR counterpart D_h. Additionally, a corresponding HR intensity image Y_h of the same scene is available. Our goal is to recover D_h using D_l and Y_h.

We first present some insights about the upsampling architecture. These motivate us on the design of our proposed upsampling CNNs.

Spectral Decomposition. We have observed that simple upsampling operator like bicubic interpolation performs very well in smooth region, but sharpness is lost along edges. Unlike SRCNN [11] and CSCN [12], we do not enlarge D_l using a *fixed* upsampling operator and then *refine* the enlarged D_l afterwards. To achieve optimal upsampling, we believe that different spectral components of D_l need to be upsampled using different strategies because a single upsampling operator is unlikely to be suitable for upsampling of all kinds of structures.

Multi-scale Upsampling. Multi-scale representation has played an important role in the success of addressing low-level problems like motion-depth fusion [27], optical flow generation [23] and depth map recovery [7]. Different structures in an image have different scales. A multi-scale upsampling CNN that allows the use of scale-dependent upsampling kernels can greatly improve the quality of the recovered HR image especially at large upscaling factors.

3.1 Formulation

We design MSG-Net to upsample a LR image D_l not in a single level but progressively in multiple levels to a desired HR image $\widetilde{D_h}$ with multi-scale guidance from the corresponding HR intensity image Y_h. We upsample D_l in m levels for the upscaling factor 2^m. Figure 3 shows an overview of the network architecture. It consists of five stages, namely feature extraction (each for Y- and D-branches), downsampling, upsampling, fusion and reconstruction. We will discuss the details of each stage in this section.

Overview. It is not possible to determine the absolute depth value of a pixel from an intensity patch alone as it is an ill-posed problem. Flat intensity patches (regardless of what intensity values they possess) do not contribute much improvement in depth super resolution. Therefore, we complement depth features with the associated intensity features in *high-frequency domain*. In other words, we perform an **early spectral decomposition** of D_l: $D_l = l(D_l)+h(D_l)$. By using the high-frequency (h) components of both Y and D images as the inputs, this gives room for the network to focus on structured features for joint upsampling and filtering. This in turn improves the upsampling performance greatly. We have also experienced a reduction in the convergence time if the network are trained in high-frequency domain. We obtain the high-frequency

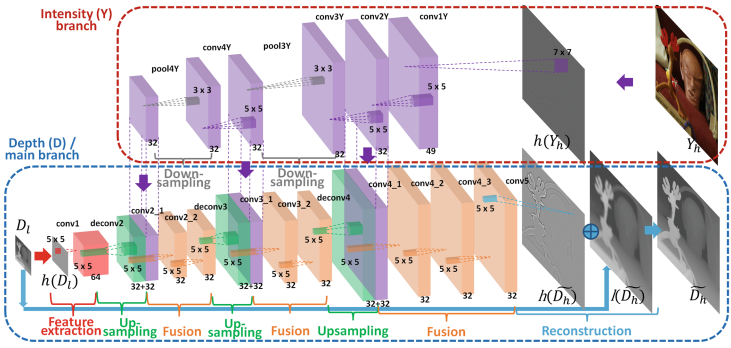


Fig. 3. The architecture of MSG-Net. For the ease of representation, only an upsampling CNN with upscaling factor 8 is presented. There are three multi-scale upsampling levels. Each level consists of an upsampling and a fusion stage.

components of D_l, D_h, and Y_h by applying a low-pass filter $\mathbf{W}_l$ to them as follows:

$$h(D_l) = D_l - \mathbf{W}_l * D_l, \tag{1.1}$$

$$h(D_h) = D_h - (\mathbf{W}_l * D_l)^{\uparrow D_h}, \tag{1.2}$$

$$h(Y_h) = Y_h - \mathbf{W}_l * Y_h, \tag{1.3}$$

where $(I_l)^{\uparrow D_h}$ performs a bicubic upsampling on I_l to the same resolution as D_h.

Suppose the upscaling factor is $s = 2^m$, then there are M layers (including m upsampling levels) in the main branch and $2m$ layers in the Y branch. MSG-Net can be expressed as follows:

$$F_1^Y = \sigma(\mathbf{W}_{c(1)}^Y * h(Y_h) + \mathbf{b}_1^Y), \text{ (feature extraction)} \tag{2.1}$$

$$F_j^Y = \sigma(\mathbf{W}_{c(j)}^Y * F_{j-1}^Y + \mathbf{b}_j^Y), \text{ (post-feature extraction)} \tag{2.2}$$

$$F_{2j'}^Y = maxpool(F_{2j'-1}^Y), \text{ (downsampling)} \tag{2.3}$$

$$F_1 = \sigma(\mathbf{W}_{c(1)} * h(D_l) + \mathbf{b}_1), \text{ (feature extraction)} \tag{2.4}$$

$$F_k = \sigma\left(\mathbf{W}_{d(k)} \star F_{k-1} + \mathbf{b}_k\right), \text{ (upsampling)} \tag{2.5}$$

$$F_{k+1} = \sigma\left(\mathbf{W}_{c(k+1)} * \left(F_{2(m+1-k/3)}^Y, F_k\right) + \mathbf{b}_{k+1}\right), \text{ (fusion)} \tag{2.6}$$

$$F_{k+2+k'} = \sigma\left(\mathbf{W}_{c(k+2+k')} * F_{k+1+k'} + \mathbf{b}_{k+2+k'}\right), k' \in \{0,1\} \text{ (post-fusion)} \tag{2.7}$$

$$F_M = h(\widetilde{D_h}) = \mathbf{W}_{c(M)} * F_{M-1} + \mathbf{b}_M, \text{ (reconstruction)} \tag{2.8}$$

$$\widetilde{D_h} = h(\widetilde{D_h}) + (\mathbf{W}_l * D_l)^{\uparrow \widetilde{D_h}}, \text{ (post-reconstruction)} \tag{2.9}$$

where $j = \{2, 3, 5, \ldots, 2m-1\}$, $j' = \{4, 6, ..., 2m\}$, $k = \{2, 5, \ldots, 3m-1\}$ and $M = 3(m+1)$. The operators $*$ and $\star$ represent convolution and backwards convolution respectively. Vectors (or blobs) having superscript Y in (2) belongs to HR intensity (Y) branch of MSG-Net. $\mathbf{W}_{c(i)/d(i)}$ is a kernel (subscripts c and

d stand for convolution and deconvolution respectively) of size $n_{i-1} \times f_i \times f_i \times n_i$ (n_{i-1} and n_i are the numbers of feature maps in the $(i-1)^{th}$ and i^{th} layers, respectively) and $\mathbf{b}_i$ is a n_i-dimensional bias vector (it is a scalar in the top layer). Each layer is followed by an activation function for non-linear mapping except the top layer. We use parametric rectified linear unit (PReLU) [28] as the activation function (σ) due to its generalization and improvement in model fitting, where $\sigma(y) = \max(0, y) + a \min(0, y)$ and a is a learnable slope coefficient for negative y.

Denote F as our overall network architecture for MSG-Net and $\Theta = \{\mathbf{W}, \mathbf{b}, \mathbf{a}\}$ as the network parameters controlling the forward process, we train our network by minimizing the mean squared error (MSE) for N training samples as follows:

$$L(\Theta) = \frac{1}{N} \sum\nolimits_{i=1}^{N} ||F(h(Y_{h(i)}), h(D_{l(i)}); \Theta) - h(D_{h(i)})||^2. \tag{3}$$

The loss is minimized using stochastic gradient descent.

Feature Extraction. MSG-Net first decomposes a LR high-frequency depth map $h(D_l)$ and the associated HR high-frequency image $h(Y_h)$ into different spectral components (sub-bands) at the bottom layer and the first two layers of the D- and Y-branches respectively. This facilitates the network to learn for scale-dependent and spectral-dependent upsampling operators afterwards.

Multi-scale Upsampling. We perform upsampling in m levels. Backwards convolution (or so-called deconvolution) (**deconv**) in the i^{th} layer is used to upsample the sub-bands $F_{i-1} = \{f_{(i-1,j)}, j = 1, \ldots, n_{i-1}\}$ in the $(i-1)^{th}$ layer. Each **deconv** layer has a set of trainable kernels $\mathbf{W}_{d(i)} = \{\mathbf{w}_{d(i,j)}, j = 1, \ldots, n_i\}$ such that $\mathbf{w}_{d(i,j)} = \{w_{d(i,j,k)}, k = 1, \ldots, n_{i-1}\}$ and $w_{d(i,j,k)}$ is a $f_i \times f_i$ filter. **Deconv** recovers the j^{th} HR sub-band in the i^{th} layer by utilizing the dependency across all LR sub-bands in the $(i-1)^{th}$ layer as follows:

$$f_{(i,j)} = \sum\nolimits_{k=1}^{n_{i-1}} w_{d(i,j,k)} \star f_{(i-1,k)} + b_{(i,j)}. \tag{4}$$

More specifically, each element in a HR sub-band is constructed by element-wise summation of a corresponding set of enlarged blocks of pixels across all the LR sub-bands in the previous layer. Suppose a stride s is used, each enlarged block of pixels is centered in a 2D regular grid with length s.

Fischer *et al.* [25] and Long *et al.* [23] proposed to feed-forward and concatenate feature maps from lower layers. MSG-Net uses a more effective design. We directly enlarge feature maps which originate from the previous layer without feed-forwarding. Unlike the "unpooling + convolution" (**uconv**) layer introduced by Dosovitskiy *et al.* [24], our upsampling uses backwards convolution in which it diffuses a set of feature maps to another set of larger feature maps. The diffusion is governed by the learned **deconv** filters but not simply filling zeros. More importantly, **uconv**s are used in their networks to facilitate the transformation from a high-level representation generated by multiple fully-connected (FC) layers to two images but not to upsample a given LR image.

To compromise both computational efficiency and upsampling accuracy, we set f_i for $\mathbf{W}_{d(i)}$ to be $2s+1$. Having such a kernel size ensures that all the inter-pixels between the demultiplexed pixels in each feature map are completely covered by **deconv** filter $\mathbf{W}_d$. We observed that $\mathbf{W}_d$ with a size larger than $(2s+1) \times (2s+1)$ does not bring significant improvement.

Downsampling. The associated HR intensity image Y_h posses the same resolution as HR depth map D_h. In our design, D_l is progressively upsampled by a factor of 2 in a multi-scale manner. In order to match the size of the feature maps for D and Y, we progressively downsample the feature maps extracted from $h(Y_h)$ in the reverse pace by a convolution followed by a 3×3 maximum pooling with stride $= 2$. Downsampling of feature maps in Y-branch can also be achieved by using a 3×3 convolution with stride $= 2$. The resulting CNN performs slightly poorer than the one using pooling.

Fusion. The upsampled feature maps F_k are complemented with the corresponding feature maps $F^Y_{2(m+1-k/3)}$ in Y-branch possessing the same resolution. The fusion kernel $\mathbf{W}_{c(k+1)}$ in (2.6) constructs a new set of sub-bands by fusing the local features in the vicinity defined by $\mathbf{W}_{c(k+1)}$ across all the sub-bands of F_i and $F^Y_{2(m+1-k/3)}$. As intensity features in Y_h may not be consistent with depth structures in D_h, a post-fusion layer is introduced to learn a better coupling. An extra post-fusion layer is included for an enhanced fusion before reconstruction.

Reconstruction. The enlarged feature maps from the previous upsampling levels are generally "dense" in nature. Due to spectral decomposition, the energy (i.e. intensity) of each pixel in an image is distributed across different spectral components. Reconstruction layer combines n_{M-1} upsampled sub-bands and recovers a HR image. Finally, we convert the recovered HR $h(\widetilde{D_h})$ from high-frequency domain back to an ordinary HR depth map $\widetilde{D_h}$ by a post-reconstruction step in (2.9). This is achieved by using the upsampled low-frequency image $(\mathbf{W}_l * D_l)^{\uparrow \widetilde{D_h}}$ in (1.2) as the missed low-frequency component for $\widetilde{D_h}$.

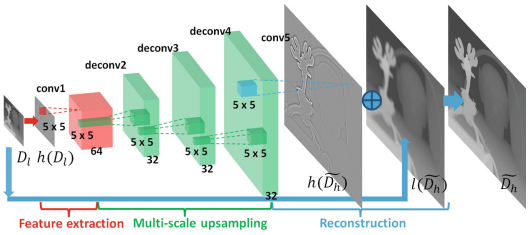


Fig. 4. The network architecture of MS-Net for single-image super resolution. For the ease of representation, only a 8× upsampling CNN is presented.

3.2 A Special Case: Single-Image Upsampling

Removing the (intensity) guidance branch and fusion stages of MSG-Net, it reduces to a compact multi-scale network (MS-Net) for super-resolving images by sacrificing some upsampling accuracy. Figure 4 illustrates its network architecture. MS-Net is used for single-image super resolution. It consists of three stages, namely feature extraction, multi-scale upsampling and reconstruction. For an upscaling factor $s = 2^m$, there are only $(m + 2)$ layers. MS-Net can be expressed as follows:

$$F_1 = \sigma(\mathbf{W}_{c(1)} * h(D_l) + \mathbf{b}_1), \text{ (feature extraction)} \tag{5.1}$$

$$F_i = \sigma(\mathbf{W}_{d(i)} \star F_{i-1} + \mathbf{b}_i), i = 2, ..., M-1, \text{ (upsampling)} \tag{5.2}$$

$$F_M = h(\widetilde{D_h}) = \mathbf{W}_{c(M)} * F_{M-1} + b_M, \text{ (reconstruction)} \tag{5.3}$$

$$\widetilde{D_h} = h(\widetilde{D_h}) + (\mathbf{W}_l * D_l)^{\uparrow \widetilde{D_h}}. \text{ (post-reconstruction)} \tag{5.4}$$

Denote F as our overall network architecture for MS-Net and $\Theta = \{\mathbf{W}, \mathbf{b}, \mathbf{a}\}$ as the network parameters controlling the forward process, we train our network by minimizing the mean squared error (MSE) for N training samples as follows:

$$L(\Theta) = \frac{1}{N} \sum_{i=1}^{N} ||F\left(h\left(D_{l(i)}\right); \Theta\right) - h\left(D_{h(i)}\right)||^2. \tag{6}$$

The loss is also minimized using stochastic gradient descent.

SS-Net vs MS-Net: Comparing the number of **deconv** parameters in the network using a single large-stride **deconv** layer (SS-Net) with that in a multi-scale small-stride **deconv** network (MS-Net), the number of **deconv** parameters for the latter one is indeed lower. Suppose all **deconv** layers in MS-Net have $s = 2$, then there are only $25 \sum_{i=2}^{m+1} n_{i-1} n_i$ kernel parameters. If they all have the same number of feature maps i.e. $n_1 = n_2 = ... = n$, then there are $25\, mn^2$ kernel parameters. For SS-Net, there are $(2^{m+1} + 1)^2 n^2$ kernel parameters.

4 Experiments

4.1 Training Details

We collected 58 RGBD images from MPI Sintel depth dataset [29], and 34 RGBD images (6, 10 and 18 images are from 2001, 2006 and 2014 datasets respectively) from Middlebury dataset [30–32]. We used 82 images for training and 10 images for validation. We augmented the training data by a 90°-rotation. The training and testing RGBD data were normalized to the range $[0, 1]$.

Instead of using large-size images for training, sub-images were generated from them by dividing each image into a regular grid of small overlapping patches. This training approach does not reduce the performance of CNN but it leads to a reduction in training time [23]. We performed a regular sampling on

the raw images with stride $= \{22, 21, 20, 24^2\}$ for the scale $= \{2, 4, 8, 16\}$ respectively. We excluded patches without depth information due to occlusion. There were roughly $190,000$ training sub-images. To synthesize LR depth samples $\{D_l\}$, we first filtered each full-resolution sub-image by a 2D Gaussian kernel and then downsampled it by the given scaling factor. The LR/HR patches $\{D_l\}/\{D_h\}$ (and $\{Y_h\}$) were prepared to have sizes $20^2/39^2$, $16^2/63^2$, $12^2/95^2$, $8^2/127^2$ for the upscaling factors $2, 4, 8, 16$, respectively. We do not prefer to use a set of large-size sub-images for training upsampling networks with large upscaling factors (e.g. $8\times, 16\times$). We have experienced that using them cannot improve the training accuracy significantly. Moreover, this increases the computation time and memory burden for training.

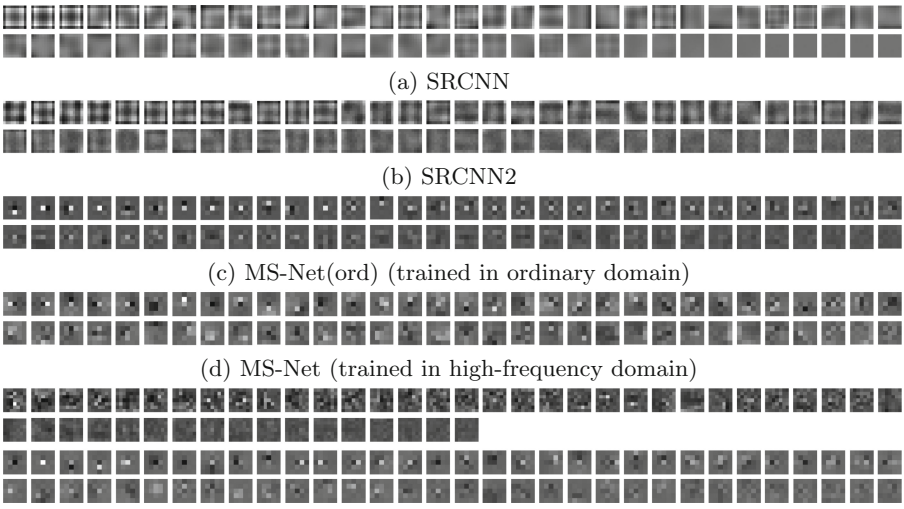
It is possible to train MS-Net (but not MSG-Net) without padding as SRCNN [11] to reduce memory usage and training time. We have to pad zeros for convolution layers in MSG-Net so that the dimension of the feature maps in the intensity branch can match that in the depth branch. We need to crop the resulted feature maps after performing backwards convolution so that the reconstructed HR depth map $\widetilde{D_h}$ is close to the desired resolution[3]. For consistency, we trained all our CNNs except SRCNN and its variant with a padding scheme.

We built our networks on top of the *caffe* CNN implementation [33]. CNNs were trained with smaller base learning rates for large upscaling factors. Base learning rates varied from 3e−3 to 6e−5 for MSG-Net and 4e−3 to 4e−4 for MS-Net. We chose momentum to be 0.9. Unlike SRCNN [11], we used stepwise decrease (5 steps with learning rate multiplier $\gamma = 0.8$) as the learning policy because we experienced that a lower learning rate usage in the later part of training process can reduce fluctuation in the convergence curve. We trained each MS-Net and MSG-Net for 5e+5 iterations. We set the network parameters: $\mathbf{W}_l = \frac{1}{9} I_3$, $f_1^Y = 7, n_1^Y = 49, n_1 = 64$ and $(f_i = 5,\ n_i = 32)$ for other layers. We initialized all the filter weights and bias values as PReLU networks [28].

We trained a specific network for each upscaling factor $s \in \{2, 4, 8, 16\}$. We adopted the following pre-training and fine-tuning scheme for MSG-Net: (1) we pre-trained the Y- and D- branches for a $2\times$ MSG-Net separately, (2) we transfered the first two layers of them (D-branch: {conv1, deconv2} and Y-branch: {conv1Y, conv2Y}) to a plain $2\times$ MSG-Net and then fine-tuned it. For training MSG-Net with other upsampling factors $(2^m, m > 1)$, we transfered all the layers except the last four layers in the D-branch from the network trained with upsampling factor 2^{m-1} to a plain network and then fine-tuned it. We trained SRCNNs for different upscaling factors using the same strategy as recommended by the authors [11]. We also modified SRCNN by replacing the activation functions from ReLU to PReLU. We name this variant as SRCNN2.

[2] For training $16\times$ MSG-Net, we reduced the amount of training samples by about 35 % using stride $= 24$ (instead of 19) in order to fulfill the blob-size limit in caffe.

[3] As we used odd-size **deconv** kernels, both the horizontal and vertical dimension of each feature map is one pixel lesser than the ideal one.



(a) SRCNN

(b) SRCNN2

(c) MS-Net(ord) (trained in ordinary domain)

(d) MS-Net (trained in high-frequency domain)

(e) MSG-Net

Fig. 5. Visualization of the bottom-layer kernels for five CNNs trained for 8× upsampling. Their kernel sizes are: 9×9 for SRCNN and SRCNN2, 5×5 for MS-Net, 7×7 (Top: Y-branch), 5×5 (Bottom: D-branch) for MSG-Net.

4.2 Analysis of the Learned Kernels

The bottom-layer filters of SRCNN which is trained for depth map upsampling are different than the one trained for image super resolution [11]. As shown in Fig. 5a, we can recognize some flattened edge-like and Laplacian filters. The filters near the right of second row are completely flat (or so-called "dead" filters). Figure 5b visualizes the filters of the trained SRCNN2. In comparison to SRCNN, SRCNN2 has sharper edge-like filters and fewer "dead" filters.

We trained MS-Net in two approaches: using ordinary and high-frequency (i.e. with early spectral decomposition) domains. As shown in Fig. 5c and d, we can recognize simple gradient operators such as horizontal, vertical and diagonal filters for both of the cases. When MS-Net is trained in ordinary domain, it first decomposes the components of LR depth map into a complete spectrum and performs spectral upsampling subsequently. By training MS-Net in high-frequency domain, all the bottom-layer kernels become high-pass filters. Similar patterned filters (bottom of Fig. 5e) are present in the first layer of the D-branch of MSG-Net as well. For the Y-branch, the learned filters (top of Fig. 5e) contain both textured and low-varying filters.

4.3 Results

We provide both quantitative and qualitative evaluations on our image-guided upsampling CNN (MSG-Net) and single-image upsampling CNN (MS-Net) to the state-of-the-art methods. We report upsampling performance in terms of

Table 1. Quantitative comparison (in RMSE) on dataset *A*.

	Art				Books				Moebius			
	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
Bilinear	2.834	4.147	5.995	8.928	1.119	1.673	2.394	3.525	1.016	1.499	2.198	3.179
MRFs [15]	3.119	3.794	5.503	8.657	1.205	1.546	2.209	3.400	1.187	1.439	2.054	3.078
Bilateral [13]	4.066	4.056	4.712	8.268	1.615	1.701	1.949	3.325	1.069	1.386	1.820	2.494
Park *et al.* [2]	2.833	3.498	4.165	6.262	1.088	1.530	1.994	2.760	1.064	1.349	1.804	2.377
Guided [8]	2.934	3.788	4.974	7.876	1.162	1.572	2.097	3.186	1.095	1.434	1.878	2.851
Kiechle *et al.* [3]	1.246	2.007	3.231	5.744	0.652	0.918	1.274	1.927	0.640	0.887	1.272	2.128
Ferstl *et al.* [4]	3.032	3.785	4.787	7.102	1.290	1.603	1.992	2.941	1.129	1.458	1.914	2.630
Lu *et al.* [6]	-	-	5.798	7.648	-	-	2.728	3.549	-	-	2.422	3.118
SRCNN [11]	1.133	2.017	3.829	7.271	0.523	0.935	1.726	3.100	0.537	0.913	1.579	2.689
SRCNN2	0.902	1.874	3.704	7.309	0.464	0.846	1.591	3.123	0.454	0.864	1.482	2.679
Wang *et al.* [12]	1.670	2.525	3.957	6.226	0.668	1.098	1.646	2.428	0.641	0.979	1.459	2.202
MS-Net	0.813	1.627	2.769	5.802	0.417	0.724	1.072	1.802	0.413	0.741	1.138	1.910
MSG-Net	**0.663**	**1.474**	**2.455**	**4.574**	**0.373**	**0.667**	**1.029**	**1.601**	**0.357**	**0.661**	**1.015**	**1.633**

Table 2. Quantitative comparison (in RMSE) on dataset *B*.

	Dolls[a]				Laundry				Reindeer			
	2×	4×	8×	16×	2×	4×	8×	16×	2×	4×	8×	16×
Bicubic	0.914	1.305	1.855	2.625	1.614	2.408	3.452	5.095	1.938	2.809	3.986	5.823
Park *et al.* [2]	0.963	1.301	1.745	2.412	1.552	2.132	2.770	4.158	1.834	2.407	2.987	4.294
Aodha *et al.*	-	1.977	-	-	-	2.969	-	-	-	3.178	-	-
CLMF0 [34]	0.990	1.271	1.878	2.291	1.689	2.312	3.084	4.312	1.955	2.690	3.417	4.674
CLMF1 [34]	0.972	1.267	1.707	2.232	1.689	2.512	2.892	4.302	1.948	2.699	3.331	4.774
Ferstl *et al.* [4]	1.118	1.355	1.859	3.574	1.989	2.511	3.757	6.407	2.407	2.712	3.789	7.271
Kiechle *et al.* [3]	0.696	0.921	1.259	1.736	0.746	1.212	2.077	3.621	0.920	1.559	2.583	4.644
AP [5]	1.147	1.350	1.646	2.323	1.715	2.255	2.848	4.656	1.803	2.431	2.949	4.088
SRCNN [11]	0.581	0.946	1.518	2.445	0.635	1.176	2.430	4.579	0.765	1.499	2.864	5.249
SRCNN2	0.473	0.881	1.461	2.422	0.506	1.084	2.314	4.601	0.603	1.352	2.740	5.330
Wang *et al.* [12]	0.670	0.989	1.445	2.107	1.039	1.630	2.466	3.834	1.252	1.914	2.878	4.526
MS-Net	0.437	0.740	1.166	1.832	0.475	0.883	1.618	3.385	0.556	1.107	1.972	3.921
MSG-Net	**0.345**	**0.690**	**1.051**	**1.597**	**0.371**	**0.787**	**1.514**	**2.629**	**0.424**	**0.984**	**1.757**	**2.919**

[a]We excluded 9 pixels for calculating RMSE as they are not filled in the ground truth.

root mean squared error (RMSE). We evaluate our methods on the hole-filled Middlebury RGBD datasets. We denote them as *A* [4], *B* [5] and *C* [19]. The RMSE values in Tables[4] 1, 2 and 3 for the compared methods are computed using the upsampled depth maps provided by Ferstl *et al.* [4], Yang *et al.* [5] and Ferstl *et al.* [19] respectively, except the evaluations for Kiechle *et al.* [3] and Wang *et al.* [12] (code packages provided by the authors), Lu *et al.* [6] (upsampled depth maps provided by the authors) and SRCNN(2) (trained by ourself). The best RMSE for each evaluation is in bold, whereas the second best one is underlined. Since the ground-truths are quantized to 8-bit, we convert all recovered HR

[4] Evaluations of several upscaling factors are not available from the authors.

Table 3. Quantitative comparison (in RMSE) on dataset *C*.

	Tsukuba			Venus			Teddy			Cones		
	2×	4×	8×	2×	4×	8×	2×	4×	8×	2×	4×	8×
Park *et al.* [2]	6.61	9.75	15.1	1.27	1.8	2.99	3.73	4.89	7.15	4.0	5.64	7.73
Li *et al.* [20]	8.29	11.9	15.84	2.29	3.55	5.76	2.78	4.92	7.24	3.24	6.34	8.9
Ferstl *et al.* [4]	7.2	10.3	17.2	2.15	2.52	4.04	2.71	3.3	5.39	3.5	4.45	7.14
Kiechle *et al.* [3]	3.48	5.95	10.9	0.8	1.17	1.76	1.28	2.94	2.76	1.7	4.17	5.11
Kwon *et al.* [7][a]	2.31	5.56	5.67	0.53	1.14	1.68	0.83	1.80	**2.19**	0.92	2.13	**2.37**
MSG-Net[b]	**1.143**	**2.233**	**3.649**	**0.142**	**0.329**	**0.762**	**0.695**	**1.307**	2.275	**0.807**	**1.772**	2.748
Aodha *et al.* [16]	8.993	12.39	-	2.175	2.597	-	3.233	4.030	-	4.262	5.740	-
Timofte *et al.* [18]	9.135	12.09	-	2.099	2.331	-	3.253	3.718	-	4.257	5.490	-
Kiechle *et al.* [3]	3.653	6.212	10.08	0.607	0.819	1.169	1.198	1.822	2.370	1.465	2.974	4.516
Ferstl *et al.* [19]	5.254	7.352	-	1.108	1.742	-	1.694	2.595	-	2.185	3.498	-
Lu *et al.* [6]	-	10.29	13.77	-	1.734	2.134	-	2.723	3.468	-	3.985	5.344
SRCNN [11]	3.275	7.939	11.28	0.456	0.789	1.706	1.170	1.985	3.252	1.484	3.585	5.180
SRCNN2	2.796	7.178	11.20	0.315	0.718	1.593	0.947	1.891	3.136	1.183	3.439	5.171
Wang *et al.* [12]	3.979	6.281	9.589	0.828	1.191	1.786	1.368	2.026	3.015	1.856	3.078	4.865
MS-Net	2.472	4.996	9.986	0.259	0.422	0.881	0.822	1.533	2.874	1.100	2.770	5.217
MSG-Net	**1.848**	**4.292**	**8.428**	**0.142**	**0.346**	**1.040**	**0.713**	**1.485**	**2.760**	**0.905**	**2.595**	**4.229**

[a]The reported values in the top-half of Table 3 are obtained from their supplementary material. Please note that depth maps for [7] are initialized using Park *et al.* [2].
[b]We used the RMSE calculation suggested by [7]: (1) Depth maps are normalized, (2) compute the absolute difference and convert it to `uint8` and (3) calculate RMSE.

depth maps in the same data type in order to have a fair evaluation. Following [6,7,19], we performed evaluation on dataset *C* only up to 8× due to the low resolution ($< 450 \times 375$) of the ground-truths.

As shown in the three tables, our single-image upsampling CNN (MS-Net) achieves state-of-the-art performance. SRCNN2 performs better than the original SRCNN due to the use of PReLU as the activation function. Although MS-Net and SRCNN(2) are both designed for single-image super resolution, MS-Net outperforms SRCNN(2). This is because MS-Net performs image upsampling but not image refinement as SRCNN(2). MS-Net (and also MSG-Net) are trained to learn different upsampling operators for different spectral components of LR depth map. They are not constrained only to a fixed non-trainable upsampling operator. The upsampling performance is further improved when MSG-Net upsamples LR depth map with the guidance from HR intensity image of the same scene. This in turn allows MSG-Net to outperform MS-Net. Figure 6 shows 8× upsampled depth maps for different methods. It is observed that HR depth boundaries reconstructed by MSG-Net are sharper than the compared methods. The evaluations suggest that multi-scale guidance has played an important role in the success of depth map super resolution in MSG-Net.

The Role of Guidance. We evaluate several variants of MSG-Net at upscaling factor 8: (1) MS(woG)-Net (without Y-branch), (2) MSG$(2, 4)$-Nets (Intensity-guidance only applied at deconv$(2, 4)$ respectively) and (3) MSG-Net(ord)

Fig. 6. Upsampled depth maps for dataset *A*. (a) Color image and ground-truth depth patches. Upsampled results from (b) Ferstl *et al.* [4], (c) Kiechle *et al.* [3], (d) SRCNN [11], and (e) MSG-Net.

(trained in ordinary domain). As summarized in Table 4, MSG-Net outperforms the others. Comparing to the partially guided variants MSG$(2, 4)$-Nets, MS(woG)-Net loses some upsampling performance due to the absence of guidance branch.

Table 4. RMSE for different variants of MSG-Net with upscaling factor 8.

	Art	Reindeer	Cones
MS(woG)-Net	2.596	1.801	4.667
MSG(2)-Net	2.510	1.866	4.514
MSG(4)-Net	2.574	1.788	4.249
MSG-Net(ord)	3.110	2.386	5.105
SSG-Net	2.770	1.954	4.517
MSG-Net	**2.455**	**1.757**	**4.229**

Table 5. Computation time (sec).

	2×	4×	8×	16×
MS-Net	0.211	0.221	0.247	0.277
MSG-Net	0.247	0.296	0.326	0.368

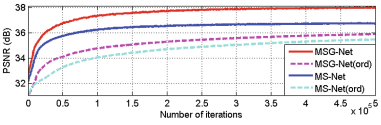


Fig. 7. Convergence curves.

The Role of Multi-scale Upsampling. We consider the single-scale variant of MSG-Net: SSG-Net (deconv4 uses stride = 8, conv3Y - pool4Y in Y-branch and deconv2 - conv3_2 in D-branch are removed). As shown in Table 4, SSG-Net performs poorer than MSG-Net. This suggests that multi-scale architecture is necessary in guided upsampling.

Training in Frequency-Domain. As presented in Table 4 and Fig. 7, MSG-Net not only performs better than its ordinary-domain trained counterpart MSG-Net(ord) in upsampling accuracy but it also converges faster. The difference in the speed of convergence is more obvious between MS-Net and MS-Net(ord). This verifies our motivation in earlier section that using high-frequency domain can facilitate depth-intensity fusion and reduce training time.

Timings. We summarize the computation time for upscaling different LR depth maps *Art* to their full resolution (1376×1088) using MS-Net and MSG-Net in Table 5. Upsamplings were performed in MATLAB with a TITAN X GPU.

5 Conclusion

We have presented a new framework to address the problem of depth map upsampling by using a multi-scale guided convolutional neural network (MSG-Net). A LR depth map is progressively upsampled with the guidance of the associated HR intensity image. Using such a design, MSG-Net achieves state-of-the-art performance for super-resolving depth maps. We have also studied a special case of it for multi-scale single-image super resolution (MS-Net) without guidance. Although sacrificing some upsampling performance, MS-Net in turn has a compact network architecture and it still achieves good performance.

Acknowledgment. This work is partially supported by SenseTime Group Limited.

References

1. Kopf, J., Cohen, M., Lischinski, D., Uyttendaele, M.: Joint bilateral upsampling. ToG **26**(3), Article No. 96 (2007)
2. Park, J., Kim, H., Tai, Y.W., Brown, M., Kweon, I.: High quality depth map upsampling for 3D-TOF cameras. In: ICCV, pp. 1623–1630 (2011)
3. Kiechle, M., Hawe, S., Kleinsteuber, M.: A joint intensity and depth co-sparse analysis model for depth map super-resolution. In: ICCV, pp. 1545–1552 (2013)
4. Ferstl, D., Reinbacher, C., Ranftl, R., Rüther, M., Bischof, H.: Image guided depth upsampling using anisotropic total generalized variation. In: ICCV, pp. 993–1000 (2013)
5. Yang, J., Ye, X., Li, K., Hou, C., Wang, Y.: Color-guided depth recovery from RGB-D data using an adaptive autoregressive model. TIP **23**(8), 3962–3969 (2014)
6. Lu, J., Forsyth, D.: Sparse depth super resolution. In: CVPR, pp. 2245–2253 (2015)
7. Kwon, H., Tai, Y.W., Lin, S.: Data-driven depth map refinement via multi-scale sparse representation. In: CVPR, pp. 159–167 (2015)
8. He, K., Sun, J., Tang, X.: Guided image filtering. PAMI **35**(6), 1397–1409 (2013)

9. Hui, T.W., Ngan, K.: Depth enhancement using RGB-D guided filtering. In: ICIP, pp. 3832–3836 (2014)
10. Shen, X., Zhou, C., Xu, L., Jia, J.: Mutual-structure for joint filtering. In: ICCV, pp. 3406–3414 (2015)
11. Dong, C., Loy, C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. PAMI **38**(2), 295–307 (2015)
12. Wang, Z., Liu, D., Yang, J., Han, W., Huang, T.: Deep networks for image super-resolution with sparse prior. In: ICCV, pp. 370–378 (2015)
13. Yang, Q., Yang, R., Davis, J., Nistér, D.: Spatial-depth super resolution for range images. In: CVPR (2007)
14. Liu, M.Y., Tuzel, O., Taguchi, Y.: Joint geodesic upsampling of depth images. In: CVPR, pp. 169–176 (2013)
15. Diebel, J., Thrun, S.: An application of Markov random fields to range sensing. In: NIPS (2005)
16. Mac Aodha, O., Campbell, N.D.F., Nair, A., Brostow, G.J.: Patch based synthesis for single depth image super-resolution. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7574, pp. 71–84. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33712-3_6
17. Yang, J., Wright, J., Huang, T., Ma, Y.: Image super-resolution via sparse representation. TIP **11**(9), 2861–2873 (2010)
18. Timofte, R., Smet, V.D., Gool, L.V.: Anchored neighborhood regression for fast example-based super-resolution. In: ICCV, pp. 1920–1927 (2013)
19. Ferstl, D., Ruether, M., Bischof, H.: Variational depth superresolution using example-based edge representations. In: ICCV, pp. 513–521 (2015)
20. Li, Y., Xue, T., Sun, L., Liu, J.: Joint example-based depth map super-resolution. In: ICME, pp. 152–157 (2012)
21. Gregor, K., LeCun, Y.: Learning fast approximations of sparse coding. In: ICML, pp. 399–406 (2010)
22. Osendorfer, C., Soyer, H., Smagt, P.: Image super-resolution with fast approximate convolutional sparse coding. In: Loo, C.K., Yap, K.S., Wong, K.W., Beng Jin, A.T., Huang, K. (eds.) ICONIP 2014. LNCS, vol. 8836, pp. 250–257. Springer, Heidelberg (2014). doi:10.1007/978-3-319-12643-2_31
23. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR, pp. 3431–3440 (2015)
24. Dosovitskiy, A., Springenberg, J.T., Brox, T.: Learning to generate chairs with convolutional neural networks. In: CVPR, pp. 1538–1546 (2015)
25. Fischer, P., Dosovitskiy, A., Ilg, E., Häusser, P., Hazirbas, C., Golkov, V., Smagt, P., Cremers, D., Brox, T.: FlowNet: learning optical flow with convolutional networks. In: ICCV, pp. 2758–2766 (2015)
26. Xie, S., Tu, Z.: Holistically-nested edge detection. In: ICCV, pp. 1395–1403 (2015)
27. Hui, T.W., Ngan, K.: Motion-depth: RGB-D depth map enhancement with motion and depth in complement. In: CVPR, pp. 3962–3969 (2014)
28. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: ICCV, pp. 1026–1034 (2015)
29. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012. LNCS, vol. 7577, pp. 611–625. Springer, Heidelberg (2012). doi:10.1007/978-3-642-33783-3_44
30. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. IJCV **47**(1), 7–42 (2002)

31. Scharstein, D., Pal, C.: Learning conditional random fields for stereo. In: CVPR (2007)
32. Scharstein, D., Hirschmüller, H., Kitajima, Y., Krathwohl, G., Nešić, N., Wang, X., Westling, P.: High-resolution stereo datasets with subpixel-accurate ground truth. In: Jiang, X., Hornegger, J., Koch, R. (eds.) GCPR 2014. LNCS, vol. 8753, pp. 31–42. Springer, Heidelberg (2014). doi:10.1007/978-3-319-11752-2_3
33. Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., Darrell, T.: Caffe: convolutional architecture for fast feature embedding. arXiv preprint arXiv:1408.5093 (2014)
34. Lu, J., Shi, K., Min, D., Lin, L., Do, M.N.: Cross-based local multipoint filtering. In: CVPR, pp. 430–437 (2012)