

Learning ramp transformation for single image super-resolution



Abhishek Singh*, Narendra Ahuja

Department of Electrical and Computer Engineering, University of Illinois at Urbana-Champaign, United States

ARTICLE INFO

Article history:

Received 8 May 2014

Accepted 18 January 2015

Available online 7 February 2015

Keywords:

Super-resolution

Segmentation

Low-level vision

ABSTRACT

We propose the use of explicitly identified image structure to guide the solution of the single image super-resolution (SR) problem. We treat the image as a layout of homogeneous regions, surrounded by ramp edges of a larger contrast. Ramps are characterized by the property that any path through any ramp pixel, monotonically leading from one to the other side, has monotonically increasing (or decreasing) intensity values along it. Such a *ramp profile* thus captures the large contrast between the two homogeneous regions. In this paper, the SR problem is viewed primarily as one of super-resolving these ramps, since the relatively homogeneous interiors can be handled using simpler methods. Our approach involves learning how these ramps transform across resolutions, and applying the learnt transformations to the ramps of a test image. To obtain our final SR reconstruction, we use the transformed ramps as priors in a regularization framework, where the traditional backprojection constraint is used as the data term. As compared to conventional edge based SR methods, our approach provides three distinct advantages: (1) Conventional edge based SR methods are based on gradients, which use 2D filters with heuristically chosen parameters and these choices result in different gradient values. This sensitivity adversely affects learning gradient domain correspondences across different resolutions. We show that ramp profiles are more adaptive, stable and therefore reliable representations for learning edge transformations across resolutions. (2) Existing gradient based SR methods are often unable to sufficiently constrain the absolute brightness levels in the image. Our approach on the other hand, operates directly in the image intensity domain, enforcing sharpness as well as brightness consistency. (3) Unlike previous gradient based methods, we also explicitly incorporate dependency between closely spaced edges while learning ramp correspondences. This allows for better recovery of contrast across thin structures such as in high spatial frequency areas. We obtain results that are sharper and more faithful to the true image color, and show almost no ringing artifacts.

© 2015 Elsevier Inc. All rights reserved.

1. Introduction

Super-resolving an image entails estimating the intensities of a high resolution (HR) version of the image from the smaller number of intensities available in the given low resolution (LR) image. For even moderate upscaling factors, this problem is fundamentally very highly ill-posed. Choosing appropriate regularizers or priors for the single image super-resolution (SR) problem has therefore been a major focus of research in this area.

Priors for the SR problem can be broadly classified into *patch based* and *edge based* priors, depending on the domains in which they operate. Perhaps the simplest patch based priors are those which assume simple models for image smoothness (such as linear or cubic). Super-resolution then simply amounts to interpolation of

the patch pixels according to the chosen model to obtain the sub-pixel values [14,15,30]. However, such methods tend to produce overly smooth results, and tend to produce artifacts such as chess-board effect along edges. A popular class of priors that are aimed at preserving sharpness are those which impose constraints on the marginal distributions of filterbank responses of the image [12,17]. Studies on statistical properties of natural images have found that these distributions are well modeled as Laplacians [12] or generalized Gaussians [17]. The constraints therefore occur in terms of fits of these distribution types to the data at hand. These priors, however, are used as a global constraint over the entire image. Spatial localization is incorporated only weakly at best [18].

More recently, learning based priors have aimed at estimating the relationship between LR and HR patches using a training database [9,28,27,22,29]. These learnt mappings are then used to predict the HR version of each patch of the given LR image. Freeman et al. [9,10] use a Markov random field (MRF) model to learn the

* Corresponding author.

E-mail addresses: asingh18@illinois.edu (A. Singh), n-ahuja@illinois.edu (N. Ahuja).

relationship between LR and HR patches. Yeung et al. [29] uses ideas from manifold learning to capture this relationship, wherein the manifold of HR patches is assumed to be locally linear, and patches are expressed as a linear combinations of its neighbors. Yang et al. [27] express image patches as linear combinations of atoms from a dictionary of a fixed set of image patches. This is then extended to the case where the dictionary itself is learnt to support a sparse representation of patches [28,26].

While learning based patch priors have demonstrated success, they do have certain shortcomings. Image patches, depending on their size, can exhibit high degrees of complexity and variability and it is not clear how many samples are sufficient to adequately model the variability seen in generic images, and for effectively learning their mappings across resolutions. The choice of patch size in learning based methods is itself rather heuristic, and has a significant effect on the number of patches required for learning and on the SR result as well. Patch based approaches also tend to be susceptible to spurious artifacts near sharp edges, since patches containing sharp transitions in intensities may be difficult to model using limited number of training patches, unless a very similar patch exist in the training set.

Edge based priors attempt to overcome some of the limitations of patch based approaches described above. *Edge smoothness* priors [5,16,24] favor smooth contours (or isophotes) in the image, and are motivated by human perceptual preferences for smooth image boundaries [5,16,24]. These priors have been effective in minimizing artifacts along high contrast boundaries while producing geometrically smooth contours. However, they do not directly consider the sharpness of intensities across edges in the estimated HR image. *Edge profile* based priors address this issue by modeling 1-D edge profiles of the image and learning (using training data) how these profiles transform across resolutions [21,20,7,23,25]. In [7], statistics of 1-D edge profiles are obtained by computing moments of the profile shape, and their transformation across resolutions is learnt. The gradient profile prior (GPP) approach of [21,20], fits a generalized Gaussian distribution to edge profiles, and uses a sharpness parameter to transform them across resolutions. The aforementioned approaches try to reap advantages of 1-D modeling over 2-D patch modeling. 1-D profiles are of lower dimensionality than rectangular patches, and can be described by one [21] or a few [7] parameters.

While existing edge profile methods extract 1-D profiles, these profiles are obtained using gradients, which still involve 2-D processing using a predefined filter. Computing gradients using predefined 2-D filters requires making strict assumptions about the geometry and scale of structures being detected [1]. Any choice of filter size and coefficients of the gradient operator essentially restricts the type of structures that can be detected, in terms of their scale and geometry. Such restrictions have a particularly detrimental effect on the SR problem, wherein *all* structures in the image, irrespective of their scale/geometry, need to be upscaled by learning correspondences between LR structures and their HR counterparts. Restrictions and assumptions on the scale/geometry introduced by 2D linear pre-processing causes distortions in learning this mapping, and therefore the advantage of subsequently using 1D profiles gets diminished.

In addition, imposing priors on gradients does not impose constraints on the absolute brightness values of the image. This sometimes leads to deviations in the brightness levels of the HR image relative to the given LR image in such methods [7,21]. We show this by an example presented in Fig. 1, which shows an upsampling result obtained by the edge profile based method of [7].

In this paper we propose a new edge profile based prior for the SR problem, that overcomes some of the limitations of existing edge based methods as described above. We propose a method which avoids 2-D preprocessing for obtaining edge information.

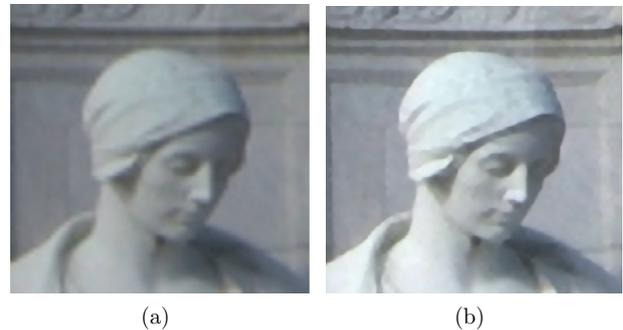


Fig. 1. (a) Bicubic upsampled image. (b) Result of edge based upsampling using [7]. Although edge sharpness is restored, the original brightness level is not maintained.

We do so by adopting the definition of image structure as proposed in [2,1]. We treat the image as a layout of homogeneous regions partitioning the image, each surrounded by ramp edges [2,1], as shown in Fig. 3(b). Such a layout is obtained by the simple requirement that the intensity variation within a region interior be strictly less than those in the ramps surrounding it. Ramps are characterized by the property that any path through a ramp pixel, monotonically leading from one to the other side, has monotonically increasing (or decreasing) intensity values along it. Such a *ramp profile* thus consists of a sharp intensity transition over a relatively narrow area between the interior intensities on the two sides, and thus captures the large contrast between the two regions.

Since ramps correspond to areas most affected by a change in resolution (as illustrated by the example in Fig. 2), we propose a prior for the SR problem that learns how ramps transform across resolutions. We model the 1-D ramp profiles using sigmoidal functions, adequately parameterized to allow the variability seen in ramp profiles extracted from natural images. We learn functions that map the ramp profiles from LR images to their HR counterparts using a set of training images. As we demonstrate in Section 2, ramp profiles are more robust descriptors of edge profiles as compared to gradient profiles. We do not use fixed size and fixed coefficient filters or templates for edge extraction. Like other edge based methods, the remaining non-ramp, homogeneous region interiors are super-resolved using the simple intensity conservation constraint [13].

Unlike gradient based edge profile priors, ramp profile modeling allows us to formulate our prior in the intensity domain. Gradient domain constraints are unable to accurately preserve brightness [7,20]. Our prior enforces sharpness directly in the image domain, thus avoiding deviations from the original intensities/colors.

Current edge profile based methods assume only a one-to-one transformation between an LR edge profile and its HR counterpart. However, we show that edge profiles across thin regions/structures also exhibit a non-trivial inter-dependency since the distance between two ramps (separated by a thin region) in an HR image may be small enough to cause an overlap between their domains of support in the corresponding LR image. We model such *inter-edge* profile relationship for better recovery of contrast across thin regions and structures.

In the next section, we describe the ramp based representation of image structure. In Section 3 we present an overview of the steps involved in our algorithm. Sections 4–7 describe our proposed algorithm in detail. Section 8 shows our results.

2. Ramp models

We now briefly review some of the key ideas of image modeling presented in [2] and the references therein.

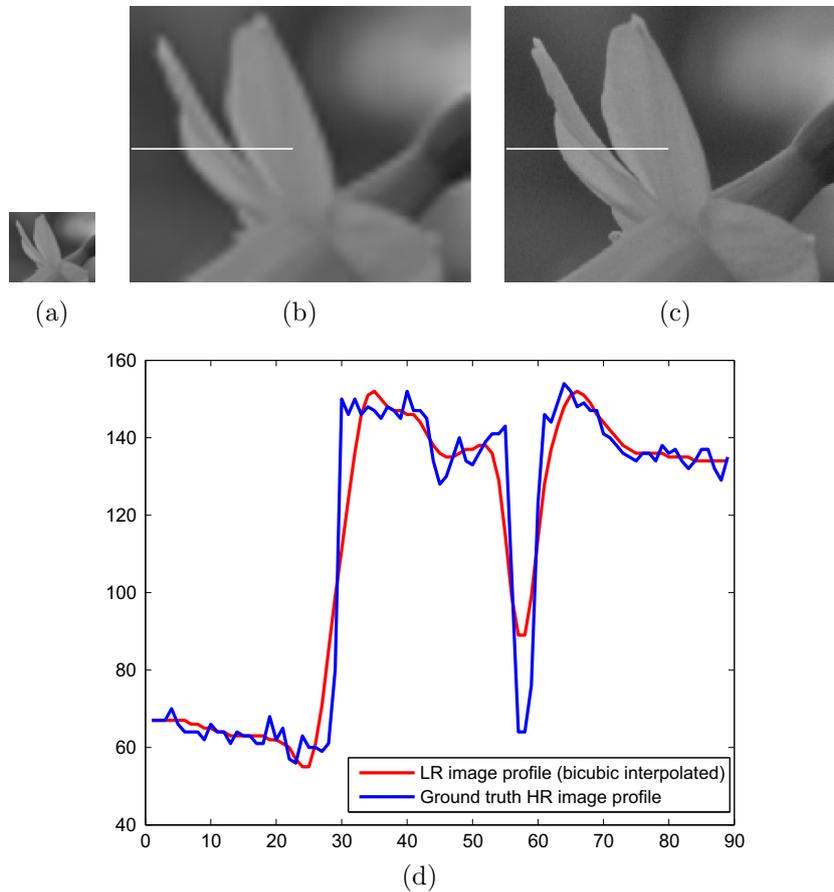


Fig. 2. (a) LR image. (b) LR image after bicubic interpolation to original image size. (c) Original ground truth HR image. (d) Intensity profiles drawn along the white line from (b) and (c). The greatest difference between the pixel profiles occurs at the ramps, which are the sharp transitions between two relatively homogenous interiors. Notice the difference in slopes of the LR and HR ramps, and also the difference in ramp heights, if two ramps are closely separated. Our algorithm learns how these ramps transform from the (bicubic interpolated) LR image to the HR image.

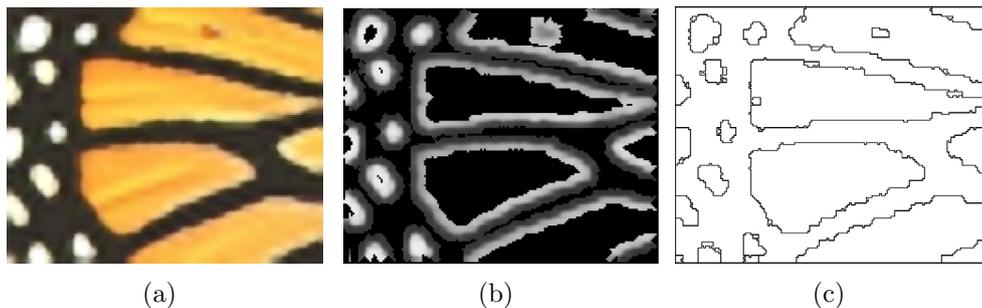


Fig. 3. (a) Original image. (b) Areas of the image containing ramps. (c) Edges contained within the ramps, like axes of the ramps.

A *ramp profile* $\mathcal{R}(\mathbf{x}, \theta)$ at a location \mathbf{x} in an image is defined as the longest sequence (ordered set) of monotonically increasing (or decreasing) pixel values along a path passing through \mathbf{x} in a particular direction θ . Ramp profiles, computed over a sufficiently large number of directions, quantify the intensity variations around a given edge location and capture the local edge structure. Ramp profiles are detected without using any predefined filters (e.g., along horizontal and vertical directions), are fully adaptive to structures of any width or height and they result from a bottom up process, without any prior assumptions [2]. We elaborate a bit more on these advantages in the context of the SR problem by a real world example in Section 3.

Using ramp profiles, Akbas and Ahuja [2] develops a low level segmentation algorithm that realizes the properties targeted in

[1]: (i) it uses a realistic model of the segments - each region having a smoothly varying interior intensity profile, surrounded by a relatively steep intensity ramps; (ii) it provides a segmentation with quantitatively and qualitatively demonstrated accuracy that does not assume any priors on region geometry (shape, size), and region topology (how many regions neighbor a given region), but rather lets the segmentation structure emerge in a bottom-up fashion; (iii) it provides regions with closed contours as well as the hierarchy of their spatial embedding, the latter not being used in this paper; (iv) the results are perceptually valid.

To summarize, for an input image, the algorithm of [2] provides us with the following: (i) A binary valued edge map E , containing closed, single-pixel wide boundaries of smooth, homogeneous regions, with $\mathcal{E} = \{\mathbf{e} : E(\mathbf{e}) = 1\}$, denoting the set of these edge

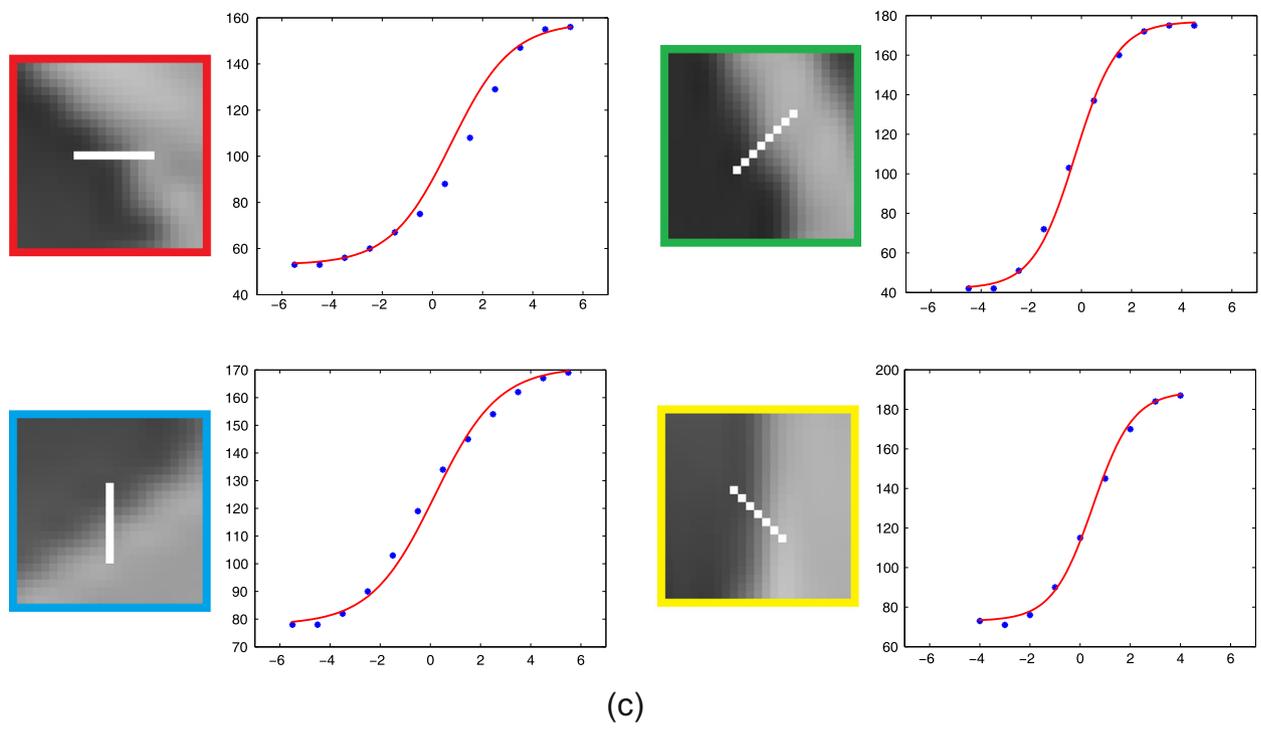
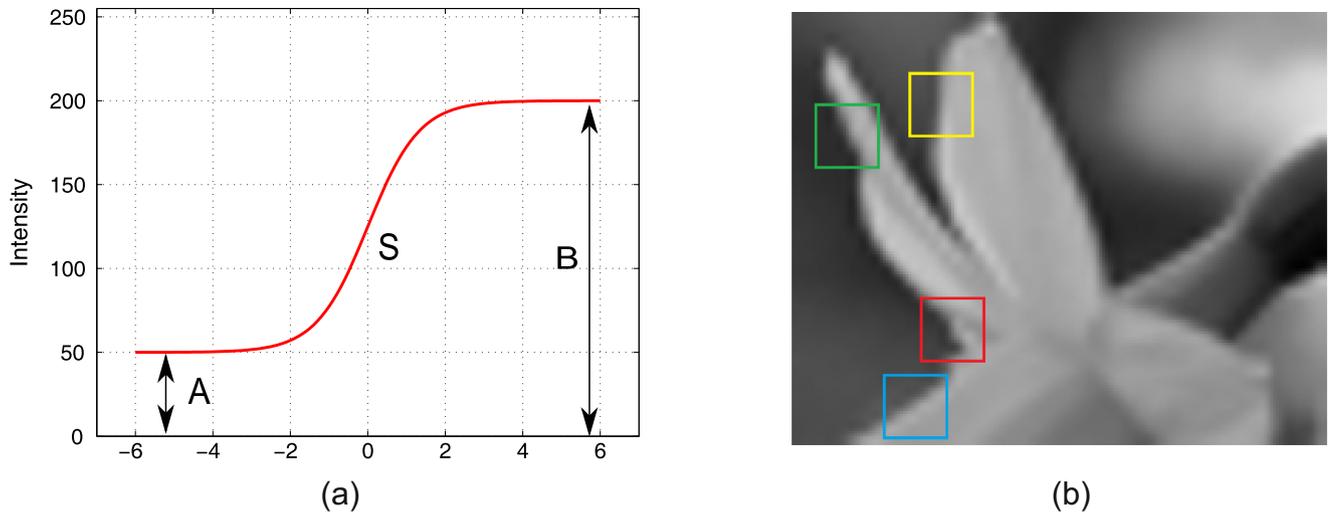


Fig. 4. (a) Parameterization of a ramp in terms of the sharpness S , and the intensity levels at either end, A and B . (b) An example image from which we extract a few sample ramp profiles and fit a sigmoid model. (c) A few ramp profiles extracted from the subimages shown in the colored boxes. The ramp profiles are denoted by the white lines in the subimages. The plots show the parametric model (red curve) fitted to the ramps extracted (blue dots) from the subimages. The sigmoid function models ramps well. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

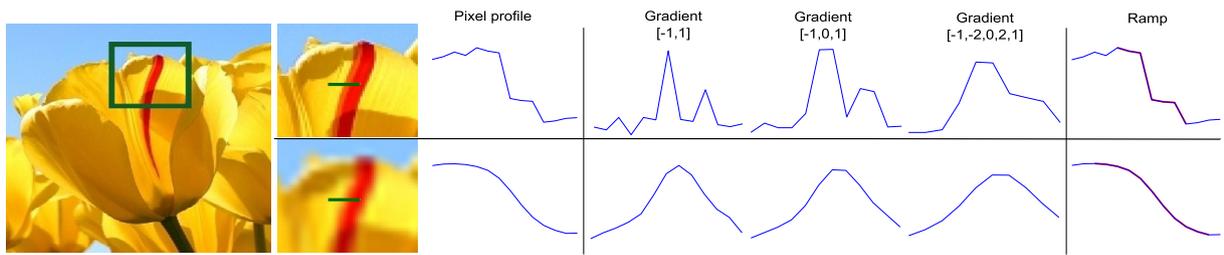


Fig. 5. We extract pixel intensity profiles from an HR image and its upscaled (bicubic) LR version, along the cross section denoted by the green line. We compute gradients along both the HR and LR profiles using various linear filters as shown. The HR gradient is quite sensitive to the filter used. Finding the location of the edge pixels, and subsequently learning correspondences between gradient profiles (as done in [21,20,7]) can be difficult given the volatility of the gradient estimation process. On the other hand, ramp profiles are able to obtain a more stable and correct localization of the edge, as shown in red in the rightmost column. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

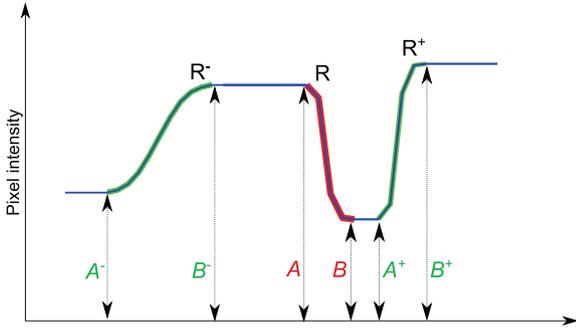


Fig. 6. Given a ramp R (in red), we denote R^- and R^+ to be the neighboring ramps (in green) on either side of the ramp R . These neighbors are detected by simply proceeding outward from either end of the ramp R , along the same cross section. We denote the heights of the ramps as the difference between the intensity levels at the ends of the ramps. Therefore, $H = A - B$, $H^- = A^- - B^-$ and $H^+ = A^+ - B^+$. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

pixels. (ii) For each edge pixel $\mathbf{e} \in \mathcal{E}$, a set of D ramp profiles $\mathcal{R}(\mathbf{e}, \theta_i), i = 1, 2, \dots, D$, along D different directions $\theta_1, \dots, \theta_D$ passing through \mathbf{e} . These ramp profiles characterize the local image structure around the edge pixel \mathbf{e} . In our work, we compute ramps along $D = 4$ different directions in the 2D image plane - horizontal, vertical and the two diagonals. Also, we deem a ramp profile valid only if it causes a specified minimum change in intensity level across it, which is above the sensor noise level. In our application, we set this threshold to be 15. Therefore, we deem a ramp profile to be valid only if it causes a gray level intensity change of at least 15 across it.

Fig. 3(c) shows the edge map obtained via such a segmentation. Fig. 3(b) shows the ramp areas, i.e., areas that are populated by the ramp profiles at the edge pixels, which can be seen as forming a thin border around the edges.

3. Overview of proposed algorithm

To super-resolve a given image I_{LR} defined in the LR domain Ω_{LR} , our algorithm consists of the following steps:

3.1. Algorithm summary

- (1) We first upscale I_{LR} to the HR domain Ω_{HR} by a simple bicubic interpolation to yield I_U .

- (2) We then use the algorithm of [2] to obtain the low-level edge map E of I_U , and the four ramp profiles $\mathcal{R}(\mathbf{e}, \theta_i), i = 1, 2, 3, 4$ for each edge pixel \mathbf{e} .
- (3) To each ramp profile, we fit a sigmoid function parameterized by Z_i , as detailed in Section 4.
- (4) We then transform Z_i to its HR counterpart Z_h , using a set of transformation functions that we learn from training images, as described in Section 5.
- (5) The transformed ramp profiles (parameterized by Z_h) are then used to create a prior image I_p in the HR domain Ω_{HR} , as described in Section 6. This prior image I_p essentially contains the ramp based structural information that the HR image is expected to have.
- (6) We estimate our final HR image \hat{I}_{HR} using a regularization framework, by using I_p as a prior constraint, along with the classical backprojection formulation [13] as the data term. This step is described in Section 7.

4. Parametric model of ramp profiles

Consider a ramp profile $\mathcal{R}(\mathbf{e}, \theta)$ that is monotonically increasing along a direction θ though the edge pixel \mathbf{e} of an image. To fit a parametric model to this ramp profile, we first consider a 1D spatial domain $t \in (-\infty, \infty)$, centered at \mathbf{e} , and along the direction θ . We can assume the ramp profile intensities to be a 1D function $r(t)$ in this domain, defined at discrete locations $t = -N, -N + 1, \dots, 0, \dots, M - 1, M$.

We model this ramp profile using a continuous sigmoid function defined as,

$$f(t; A, B, S) = A + \frac{B - A}{1 + \exp(-St)} \quad (1)$$

A and B denote the intensity values at the ends of the ramp, and $H = B - A$ denotes the height of the ramp profile. S controls the sharpness or steepness of the ramp profile. Fig. 4(a) shows these parameters schematically.

We set $A = r(-N)$, and $B = r(M)$ as the intensity levels at the end of the ramps. The least squares estimate for S can be analytically obtained as $S = [\mathbf{t}^T \mathbf{t}]^{-1} [\mathbf{t}^T \mathbf{r}]$, where $\mathbf{t} = [N, N - 1, \dots, 0, \dots, -M + 1, -M]^T$ and $\mathbf{r} = [r(-N), r(-N + 1), \dots, r(0), \dots, r(M - 1), r(M)]^T$.

Since this parameter estimation is simple and non-iterative, we are able to parameterize a large number of ramps relatively fast. We show a few examples of ramps extracted an image, superimposed with the above sigmoidal fit in Fig. 4(c).

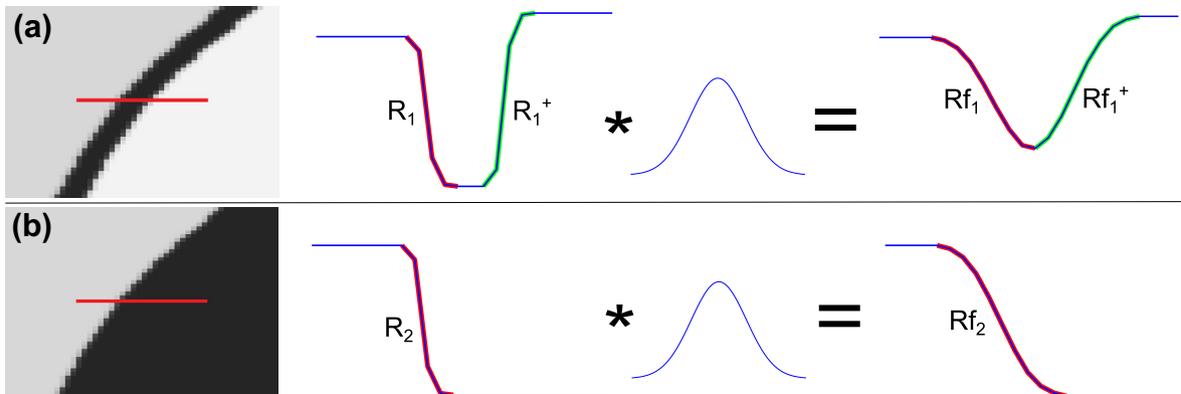


Fig. 7. Example illustrating the dependency of neighboring ramps across thin structures in the downsampling process. R_1 and R_2 are identical ramps across the red line in the two images in (a) and (b), but R_1 has a neighboring ramp separated by a thin region. After using a downsampling filter, the filtered ramps Rf_1 and Rf_2 are not identical, due to the influence of the neighboring ramp R_1^+ on R_1 during the filtering process.

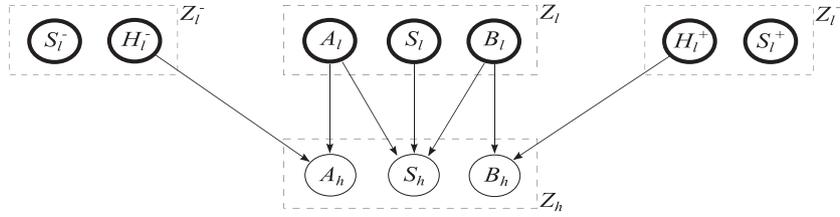
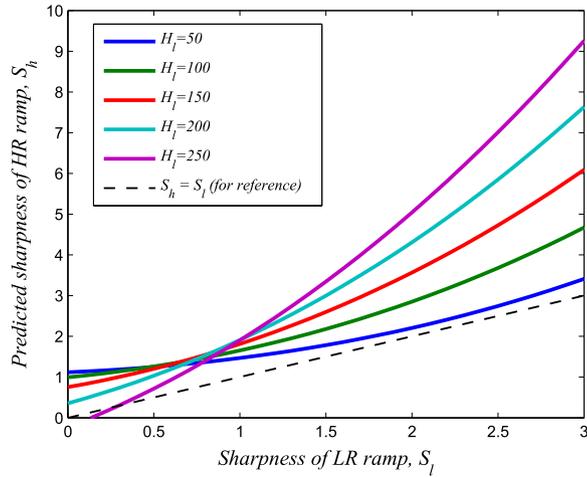
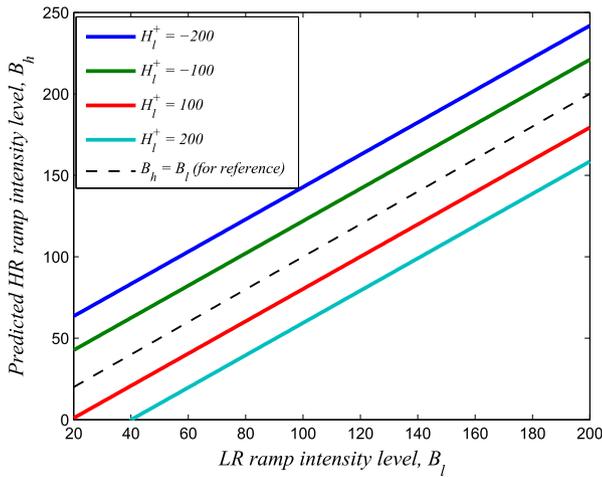


Fig. 8. Graphical model illustrating the dependency assumed among the various variables in our model. Bold circles indicate the observed variables. Thin circles denote the latent variables to be estimated.



(a)



(b)

Fig. 9. (a) The learnt HR ramp sharpness parameter S_h , shown as a function of the corresponding LR ramp sharpness S_l , at a few discretely sampled values of the LR ramp height H_l . The HR ramp sharpness depends not only on the LR ramp sharpness but also on the LR ramp height. (b) Learnt linear function for predicting B_h , as a function of B_l , for different values of H_l^+ . We show the $B_h = B_l$ line for reference. The intensity level of the HR ramp end, B_h , is clearly dependent on the neighboring LR ramp height H_l^+ .

To summarize, the above modeling procedure parameterizes the shape of a ramp profile $\mathcal{R}(\mathbf{x}, \theta)$ with a parameter vector $Z = [A, B, S]$.

5. Learning ramp transformations

In order to determine how an LR ramp profile, parameterized by Z_l , transforms to its HR counterpart Z_h , we need to learn functions

that map Z_l to Z_h using a set of known pairs of LR and HR ramp profiles $\{(Z_l^{(i)}, Z_h^{(i)})\}_{i=1}^T$. For this, we collect a set of HR images (of around 1000x1000 pixels), covering a variety of scenes, and then generate the LR images by downsampling using a filter f_{psf} .

5.1. Creating training data

For obtaining a pair $(Z_l^{(i)}, Z_h^{(i)})$, we need to extract a ramp profile from an LR image and find its corresponding ramp in the HR image. We use the segmentation algorithm [2] to obtain the edge pixels and the associated ramp profiles for all the HR images, and the LR images after upscaling (using bicubic interpolation) to the HR domain. We perform this upscaling step to have both the LR and HR image defined over the same resolution domain, as this would facilitate in finding correspondences.

For an LR ramp profile $\mathcal{R}_l(\mathbf{e}_l, \theta)$ through the edge pixel \mathbf{e}_l along direction θ in an LR image, the corresponding ramp profile $\mathcal{R}_h(\mathbf{e}_h, \theta)$ in the HR image is found at a location \mathbf{e}_h given by,

$$\mathbf{e}_h = \operatorname{argmin}_{\mathbf{e}_h \in \mathcal{N}(\mathbf{e}_l)} (|H_h(\mathbf{e}_h, \theta) - H_l(\mathbf{e}_l, \theta)|), \quad (2)$$

where $\mathcal{N}(\mathbf{e}_l)$ is the set of edge pixels of the HR image, in a 5×5 neighborhood around \mathbf{e}_l . The function $H(\mathbf{x}, \theta)$ quantifies the height of the ramp profile through the pixel \mathbf{x} , along the direction θ , in the image.

Ramp profiles allow for more accurate correspondences to be found, as compared to using gradient profiles. This is illustrated in Fig. 5. We show a cross section of an image from its HR version and its upsampled LR version. The gradient profile is quite sensitive to the type of gradient filter used, and it is difficult to infer the edge location in the HR image and establish correspondence to LR. Making any decision on the type of filter to use imposes strong assumptions of the expected geometry and scale of the image profile. On the other hand, by definition, ramp profiles avoids any such assumptions on scale, and detects structure bottom-up, adaptively. It is able to correctly identify the HR edge in the example shown in Fig. 5.

We collect T pairs of LR–HR ramp profiles from the training images, using the matching criterion in (2).

5.2. Effect of neighboring ramps

To learn a regression function from Z_l to Z_h , we also need to account for the dependency between an HR ramp profile and the closely spaced neighbors (across thin structures etc.) of the corresponding LR ramp profile. Fig. 6 shows an example of a ramp profile R (in red), along with its two neighboring ramp profiles R^- and R^+ (in green) in either direction. Without loss of generality, we denote the left neighboring ramp profile with the superscript ‘-’, and the right neighbor with a superscript ‘+’. The intensity values at the ends of the ramps are respectively denoted by A and B , with the appropriate superscript as shown in Fig. 6. Therefore, we also can denote the heights of the neighboring ramps as $H^- = B^- - A^-$ and $H^+ = B^+ - A^+$ respectively.

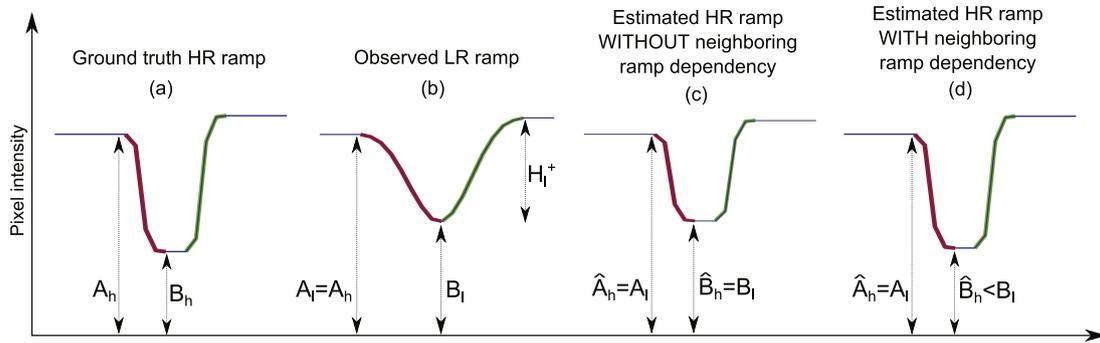


Fig. 10. 1D example illustrating the effect of our neighboring ramp dependency model. (a) Ground truth HR ramp in red, with a closely spaced neighboring ramp in green. (b) LR ramp obtained by filtering (a) with a low pass filter. Note that the height of the red ramp is significantly reduced due to the effect of the neighboring ramp, as described in the text. (c) The result of transforming the LR ramp using the proposed ramp transformation, *without* incorporating neighborhood dependency. While the sharpness of the estimated HR ramp is similar to that of the corresponding ground truth HR ramp, the height is significantly smaller. (d) The estimated HR ramp *with* the proposed neighborhood dependency model. In this case, the height of the estimated HR ramp is closer to the ground truth HR ramp, because the predicted ramp end intensity level \hat{B}_h is smaller than B_h . (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

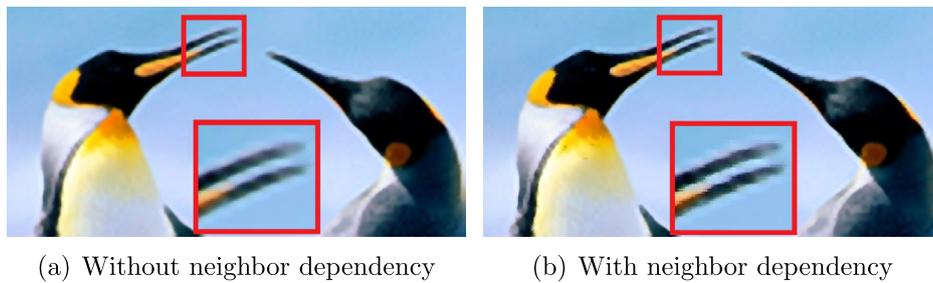


Fig. 11. Incorporating dependency between neighboring ramps across edges allows for better recovery of contrast across thin structures such as the beaks in the above images.

The distance between two neighboring ramps along the same cross section in an HR image may be small enough to cause an overlap between the spatial supports of the corresponding ramps in the LR image. This effect of this dependency is illustrated through a simple example in Fig. 7. In both cases of Fig. 7, as expected, the filtering operation causes a change in sharpness of the ramps. However, in case of Fig. 7(a), due to the presence of a close neighboring ramp, along with sharpness, the height of the filtered ramp Rf_i is also affected. The height remains unaffected if the ramp does not have neighboring ramps such as in Fig. 7(b).

To incorporate this dependency of neighboring ramps in our model, we formulate our regression function to be,

$$\hat{Z}_h = E(Z_h | Z_i, Z_i^+, Z_i^-), \quad (3)$$

where Z_i^+ and Z_i^- denote the parameters of the neighboring ramps on either side of the ramp Z_i , along the same cross section of the image. The above equation essentially says that the parameters of the HR ramp Z_h are predicted not only by its corresponding LR ramp Z_i , but also by the LR ramp's neighbors, Z_i^+ and Z_i^- if they are close

enough. We make use of the ramp map (such as in Fig. 3(b)) to determine if the ramps are close enough to require modeling using (3). Neighboring ramp profiles are deemed to be close and dependent on each other if there are no non-ramp pixels between them. In the example shown in Fig. 7(a), the neighboring LR ramps profiles Rf_i and Rf_i^+ do not have any non-ramp pixels between them. Therefore, we use the dependency model of (3) to relate them to the HR ramps profiles R_i and R_i^+ . For all other ramp profiles, we drop the dependency on Z_i^+ and Z_i^- and simply assume a one-to-one function.

5.3. Estimating prediction functions

We make some simplifying independence assumptions among the variables in (3) in order to make the estimation tractable: By comparing the filtered ramps Rf_i and Rf_j in Fig. 7, we notice that the presence of a neighboring ramp essentially reduces the height of the ramp during the filtering process. Therefore, our regression function must be aimed at compensating for this attenuation in the ramp height. Furthermore, we notice that the attenuation in height is caused by change in intensity level at only one end of

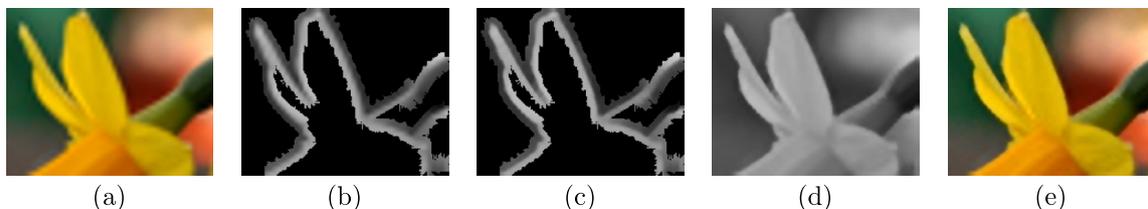


Fig. 12. (a) Bicubic upsampled LR image. (b) Ramps extracted from LR image. (c) Transformed ramps using learnt transformation functions. (d) Prior image I_p comprising of the transformed ramps and the bicubic interpolated values from (a) in the non-ramp regions. (e) HR image estimated using the prior constraint image (d) and the backprojection constraint.



Fig. 13. Training images used in our algorithm.

the ramp, that is closer to the neighboring ramp. We can therefore incorporate the neighborhood dependency by modeling A_h and B_h as functions of the neighboring LR ramp profiles, in the respective directions. Therefore,

$$\hat{A}_h = E(A_h | A_l, H_l^-) \quad (4)$$

$$\hat{B}_h = E(B_h | B_l, H_l^+) \quad (5)$$

We assume the sharpness parameter of the HR ramp profile S_h to always be independent of the neighborhood ramps. We model S_h as a function of the height and sharpness of corresponding LR ramp profile, without any neighborhood dependency.

$$\hat{S}_h = E(S_h | S_l, H_l) \quad (6)$$

Fig. 8 shows a graphical representation of the dependence and independence relationship assumed in our model.

To estimate the prediction functions (4)–(6), we take a discriminative modeling approach. We approximate $\hat{S}_h = E(S_h | S_l, H_l)$ using support vector regression with a polynomial kernel. We choose the polynomial order to be 3 based on k -fold ($k = 10$) cross validation by partitioning the training data. Fig. 9(a) shows plots of the learnt \hat{S}_h as a function of the LR sharpness S_l , for different values of LR ramp heights H_l . Clearly, there is a significant dependence of the HR ramp sharpness on not just the LR ramp sharpness, but also the LR ramp height H_l .

We use a linear model to estimate \hat{A}_h and \hat{B}_h in (4) and (5) in terms of A_l, H_l^- and B_l, H_l^+ respectively. Fig. 9(b) shows the learnt \hat{B}_h as a function of B_l , for different values of H_l^+ . \hat{A}_h behaves similarly, and we do not show it here.

Clearly, the estimation of \hat{B}_h is dependent on the neighboring ramp height H_l^+ . To better understand the plots in Fig. 9(b), let us first focus on the $\hat{B}_h = B_l$ line that is shown in the plot for reference. This line shows the case when the intensity level at the end of the HR ramp B_h is the same as the intensity level B_l at the end of its corresponding LR ramp. Indeed, this is the case if there are no neighboring ramps present. However, due to the presence of a neighboring ramp, the predicted intensity level \hat{B}_h of the HR ramp deviates from B_l . This deviation is dependent on the height H_l^+ of the neighboring ramp. For example, let's focus on the red curve, which corresponds to the presence of a neighboring ramp of height $H_l^+ = 100$. Qualitatively, we show such an example in Fig. 10 (b), where the red colored ramp is to be super-resolved, and the green

colored ramp is the neighboring ramp of height H_l^+ . Fig. 10(c) shows the transformed ramp, *without* incorporating the neighboring ramp dependency. In this case, while the sharpness of the ramp is appropriately transformed, the height of the ramp remains the same as in the LR ramp of Fig. 10(b). However, this height is lower than the ground truth HR ramp height as in Fig. 10(a). To compensate for this smaller height as compared to the ground truth, our neighboring ramp dependency model predicts a *lower* intensity value \hat{B}_h at the ramp end, as compared to the LR ramp end intensity B_l , in Fig. 10(d). This is evident by the plots of Fig. 9(b), such as the red curve. Due to this lower predicted intensity level of the ramp end, the resultant ramp in Fig. 10(d) is similar in height to the ground truth HR ramp of Fig. 10(a).

In case of real world images, the effect of incorporating our neighboring ramp dependency on our SR results is demonstrated in Fig. 11. Thin structures like the bird's beak show better contrast, owing to the correction provided by the learnt function in Fig. 9(b).

6. Ramp based prior

Given an LR image I_{LR} (and its bicubic-upscaled version I_U) to be super-resolved, each ramp profile in I_U is transformed using the prediction functions learnt above. The transformed sigmoids are then resampled at the positions where the corresponding LR ramp profiles were defined, and the intensities thus obtained are placed in a new image I_p . Let $\Omega_R \subset \Omega_{HR}$ denote the set of pixels of I_U or I_p that is populated by ramp profiles (as shown in the example of Fig. 3(b)).

Unlike previous gradient based approaches that model gradient profiles in only one direction (in the direction of the gradient), we extract and transform ramp profiles in four directions. As a result, the value $I_p(\mathbf{x})$ of a particular ramp pixel $\mathbf{x} \in \Omega_R$ is typically predicted by multiple transformed ramp profiles. We perform a weighted average of all these predictions, to get the final predicted value of $I_p(\mathbf{x})$.

$$I_p(\mathbf{x}) = \frac{\sum_j S_h(\mathbf{x}, \theta_j) H_h(\mathbf{x}, \theta_j) I_{p\theta_j}(\mathbf{x})}{\sum_j S_h(\mathbf{x}, \theta_j) H_h(\mathbf{x}, \theta_j)}, \quad \mathbf{x} \in \Omega_R. \quad (7)$$

$S_h(\mathbf{x}, \theta_j)$ and $H_h(\mathbf{x}, \theta_j)$ are the sharpness and height of the (transformed) ramp profiles through \mathbf{x} in direction θ_j , and $I_{p\theta_j}(\mathbf{x})$ is the intensity predicted at \mathbf{x} by the (transformed) ramp profile along direction θ_j alone. Intensities predicted by high contrast and sharp ramp profiles have higher weight.

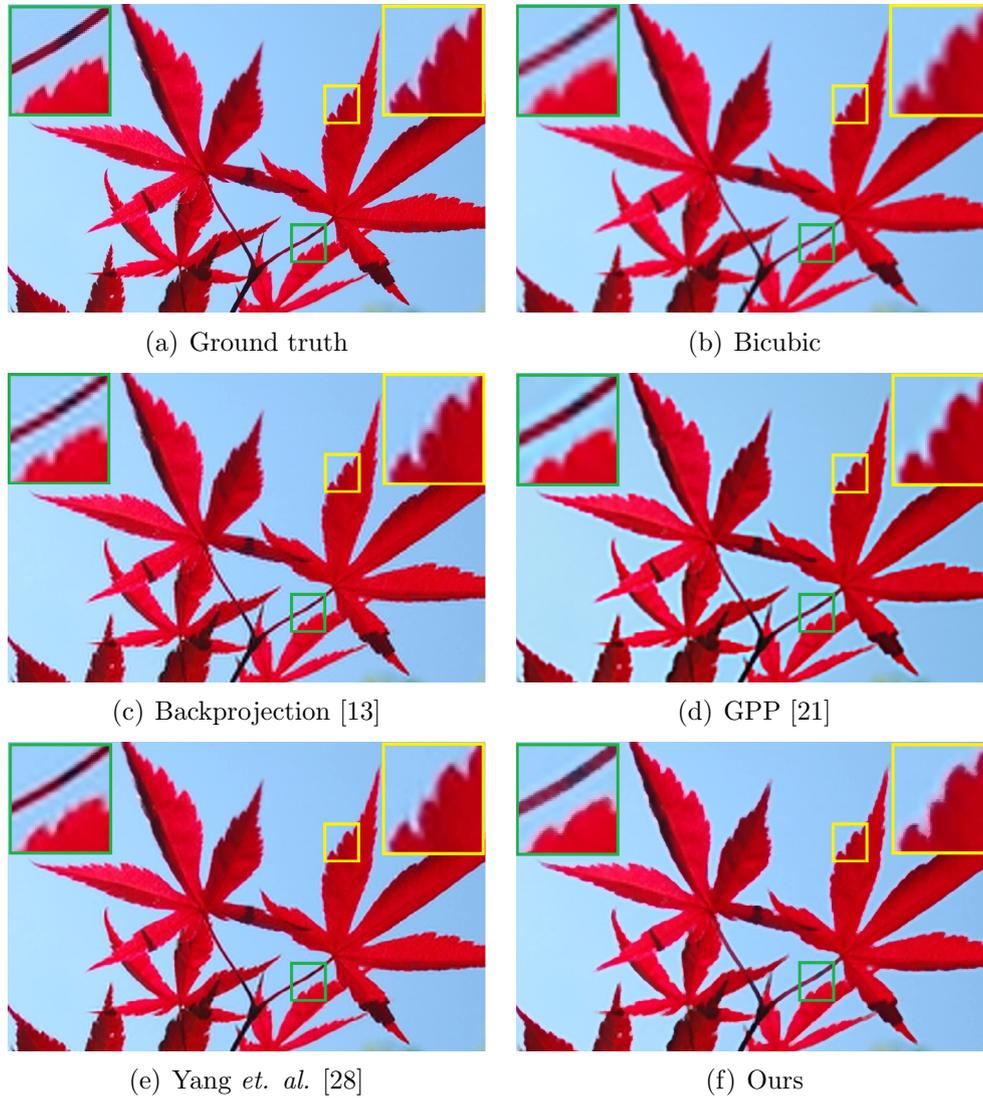


Fig. 14. Leaves (4X). Thin structures (e.g., the stem) are well reconstructed in our result, and sharpness along the leaves is also better maintained as compared to other methods.

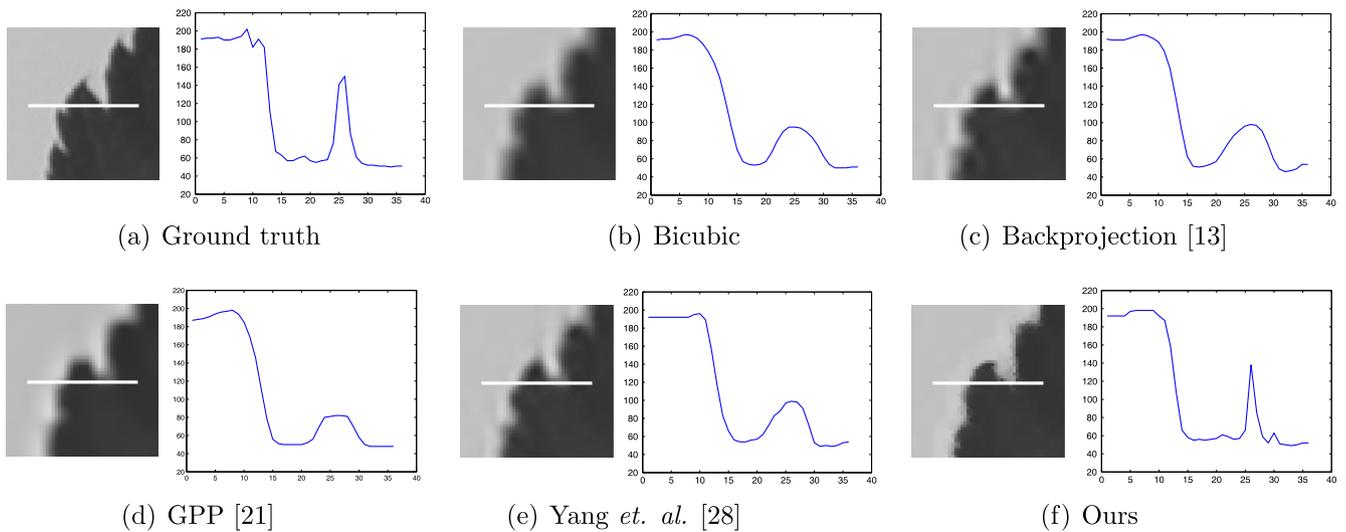


Fig. 15. Leaves 1-D profiles. We extract 1-D intensity profiles (along the white lines) from the results of the Leaves image of Fig. 14, and plot them. As evident from the plots, our algorithm is able to better reconstruct the ramp transitions, and our profile bears closer resemblance to the ground truth. We can see that thin/narrow structures are flattened out by other methods, while our result does a better job of preserving such details.

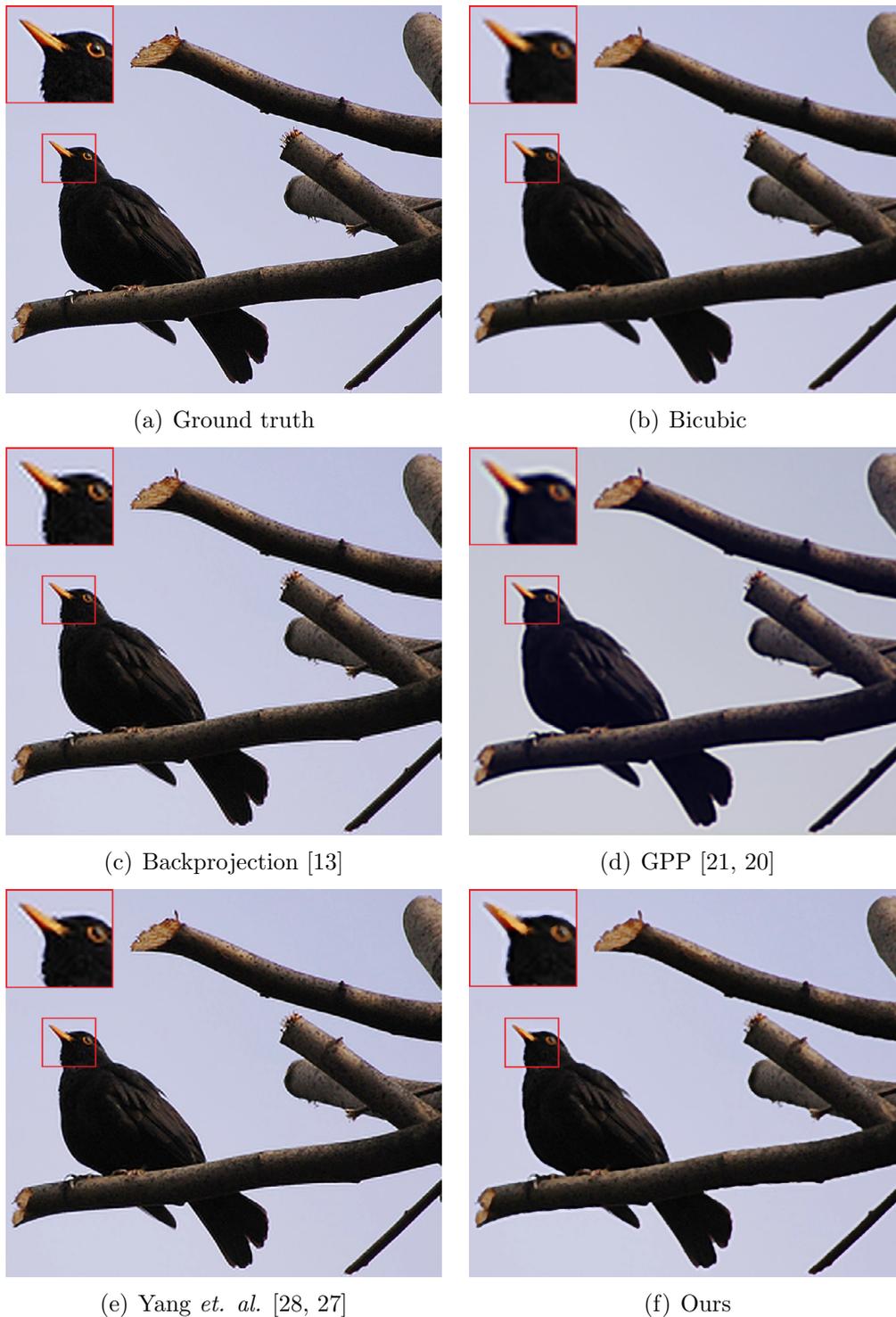


Fig. 16. Bird (4X). GPP [21] is unable to preserve the color of the sky. Yang et al. [28] exhibits ringing artifacts along the bird's beak. Our results are sharp, preserve the original image color, and show little or no ringing artifacts.

Due to this averaging, smoothness is automatically achieved between neighboring pixels in I_p , without an explicit Markov chain based inference [7,22].

For the non-ramp locations in I_p , we simply retain the bicubic interpolated values from I_U . This new image I_p serves as our ramp based prior constraint.

Fig. 12 shows an example where we extract ramps from an upsampled LR image, transform the ramps using the learnt functions, and obtain the prior image I_p .

7. SR reconstruction

We use the prior I_p together with the intensity conservation constraint in the LR domain to estimate the HR image. The cost function to minimize is therefore,

$$J(\hat{I}_{HR}) = \left\| (1 - \mathcal{A}) \downarrow \circ [(\hat{I}_{HR} * f_{psf}) \downarrow - I_{LR}] \right\|^2 + \left\| \mathcal{A} \circ (\hat{I}_{HR} - I_p) \right\|^2 \quad (8)$$

' \circ ' denotes the Hadamard (entry-wise) product between matrices. \mathcal{A} is a matrix containing spatially adaptive regularization

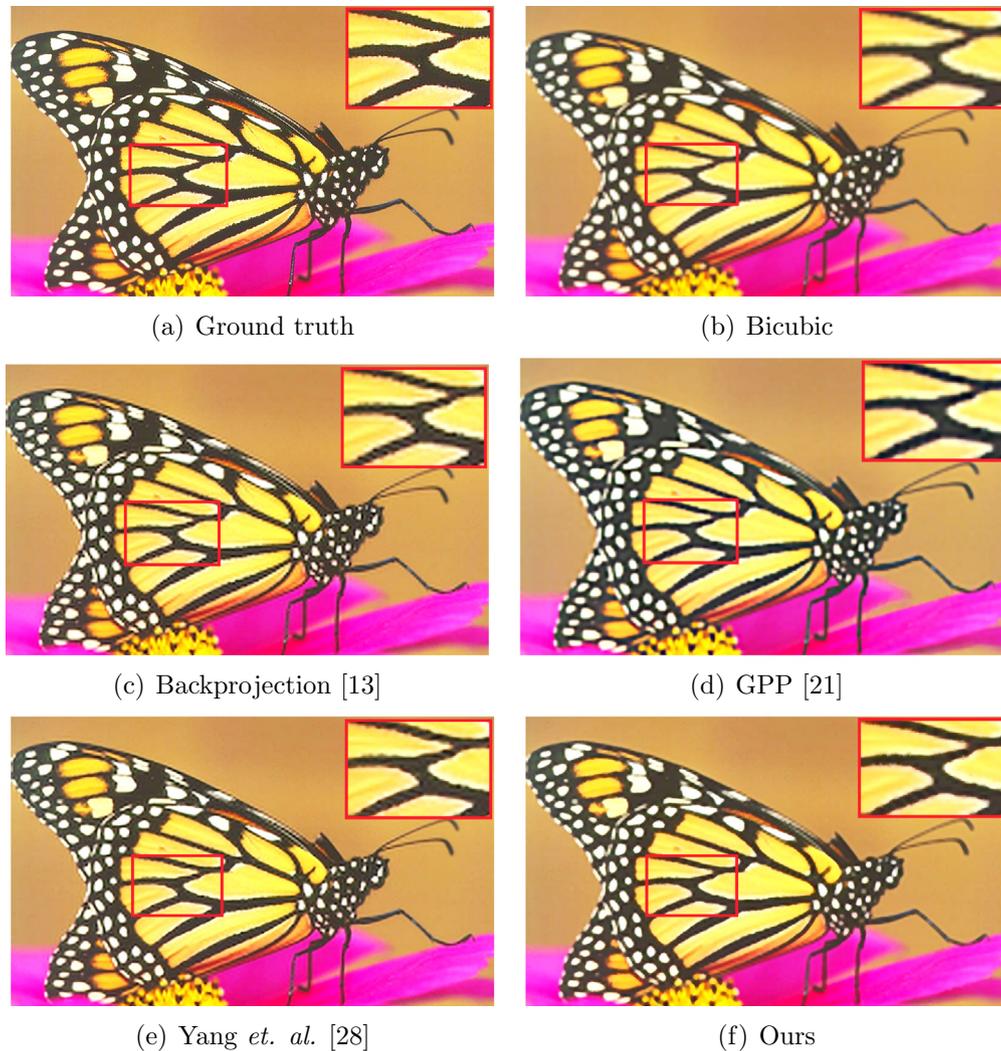


Fig. 17. *Monarch* image (4X). Yang et al. [28] (e) produces considerable ringing around high contrast edges. GPP [21] (d) is less sharp and the image color is slightly off compared to the original. Our result is closest to the original, with sharp edges and no visible ringing effect.

parameters, defined as, $\Lambda = \lambda M * g$, where $M(\mathbf{x}) = 1$ if $\mathbf{x} \in \Omega_R$, and 0 everywhere else. M is therefore a binary valued matrix used as a map to indicate the ramp regions. Λ is obtained by smoothing out the map M using Gaussian filter g and rescaling it by λ .

Λ gives high weight to our prior in the ramp areas. In smooth regions, $J(\hat{I}_{HR})$ defaults to the backprojection formulation [13]. We minimize (8) using gradient descent.

8. Experimental results

8.1. Implementation details

We used a training set of 10 LR–HR image pairs, as shown in Fig. 13. We used a 7×7 Gaussian filter of width 1.6 as f_{psf} for downsampling the HR training images by a factor of 4, to create the LR versions. The same f_{psf} is used for the reconstruction in (8) as well. We extracted around $T = 200,000$ pairs of LR–HR ramp profile pairs for learning. We found that learning with even $T = 20,000$ produced comparable results. We trained our ramp transformations for an upscaling factor of 4X. We choose $\lambda = .8$ as the regularization parameter in (8), as this yielded the most visually pleasing results. For processing color images, we apply the proposed SR method only on the luma component. The chroma channels are upscaled using bicubic interpolation.

8.2. Evaluation strategy

Despite the growing interest in the single image SR problem within the image processing and vision communities, it still lacks a common benchmark for objective evaluation and comparisons. Accurate numerical and quantitative evaluation is also an open problem. Peak signal-to-noise ratio (PSNR) and structural similarity measure (SSIM) [31] have been two commonly used numerical measures in the past. However, their correlation with human perception of image quality (which tends to be sharpness-driven) is debatable. Numerical measures like PSNR often tend to favor smoother images, whereas the challenge in the SR problem is to recover adequate sharpness. Sharp reconstructed edges yielded by SR algorithms are susceptible to high numerical errors due to the ambiguity in edge localization in the HR domain. Perhaps for this reason, several recent state-of-the-art SR methods (such as [11,8,7,19]) emphasize visual quality rather than quantitative comparisons, and provide results without any ground truth HR versions.

For evaluation, we show most of our results on images that have ground truth HR versions. For these images we report both PSNR and SSIM measures. Although we report these numbers, for better evaluation we encourage the reader to examine the visual similarity of our results with the ground truth images, as compared to competing methods. In some cases where our numerical results are close to those of others, visually our results appear better and

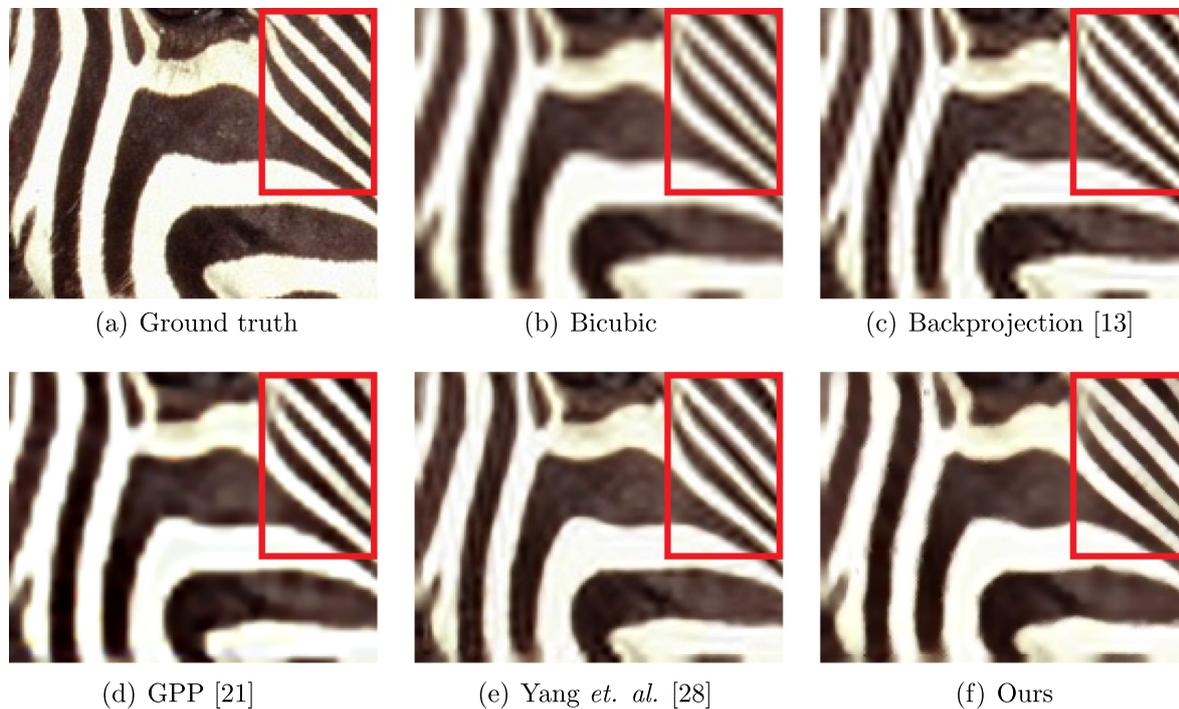


Fig. 18. Stripes (4X). The stripes in our result are sharp, without ringing effect, and bears close visual resemblance to the original image.

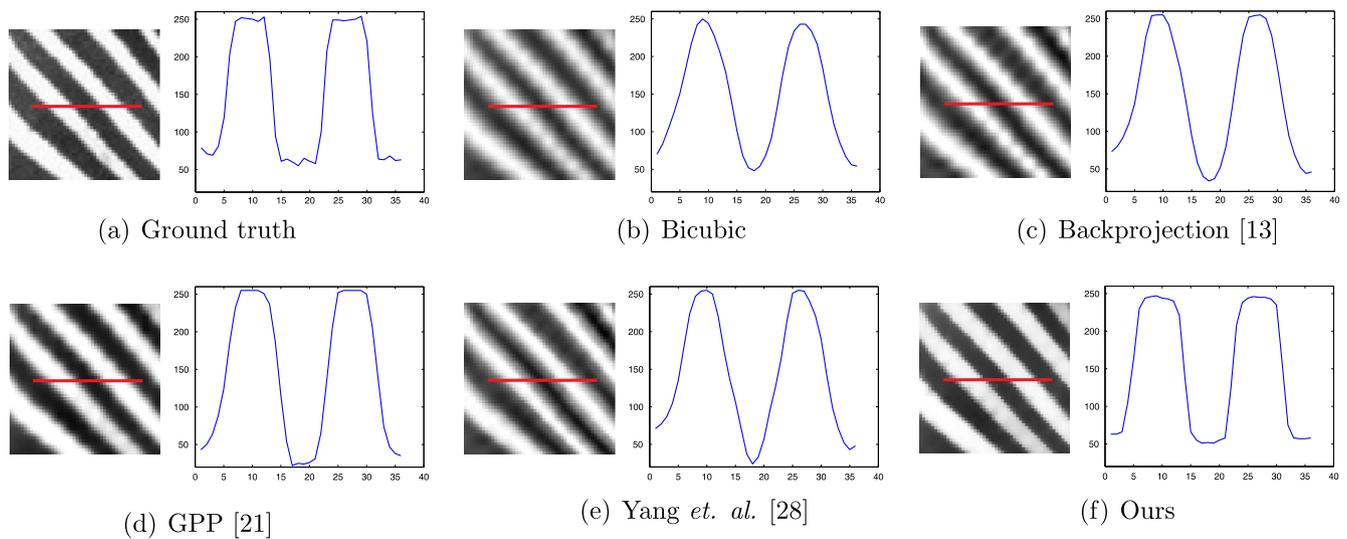


Fig. 19. Stripes 1-D profiles. We extract 1-D intensity profiles (along the red lines) from the results of the Stripes image of Fig. 18, and plot them. Our algorithm is able to produce ramps that bear closer resemblance to the ground truth. The ramp transitions produced by other methods are not as sharp. Also, the GPP algorithm [21] is not able to reproduce the correct intensity level for the black stripes.

Table 1
Comparison of PSNR (db) and structural similarity measure (SSIM) [31] for the Leaves, Bird, Monarch and Stripes images. Values in bold indicate the best performance among the different methods.

Image	Bicubic	Backproj. [13]	Yang [28]	GPP [21]	Ours
Leaves	28.88 .9197	29.78 .9148	30.94 .9316	26.98 .9164	30.96 .9344
Bird	27.29 .8677	28.08 .8718	29.43 .8917	24.16 .8364	29.84 .8938
Monarch	24.23 .8426	24.97 .8451	25.81 .8656	22.54 .8261	25.67 .8661
Stripes	18.58 .7203	19.85 .7585	20.77 .8191	16.04 .6620	20.31 .8146

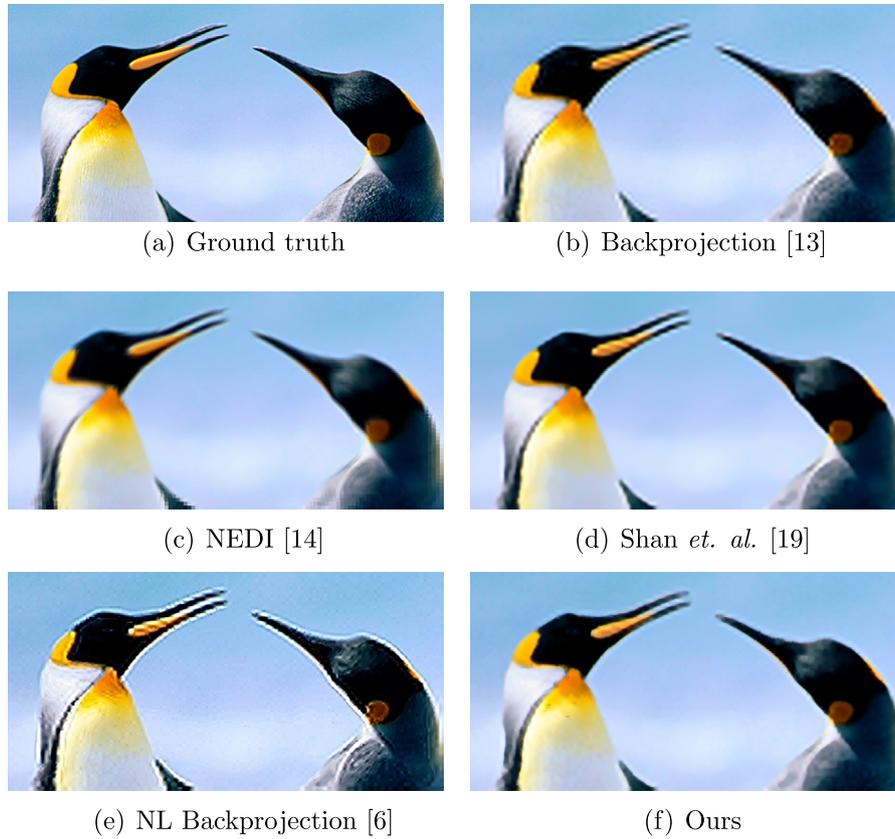


Fig. 20. Penguin image (4X). NEDI [14] tends to oversmooth the image. NL backprojection [6] shows less ringing than the classical backprojection algorithm [13], but introduces other artifacts along edges. Shan et al. [19] produces good results, but also shows some ringing artifacts along the beaks. Our result is sharp, without ringing effect, and match the original image closely.

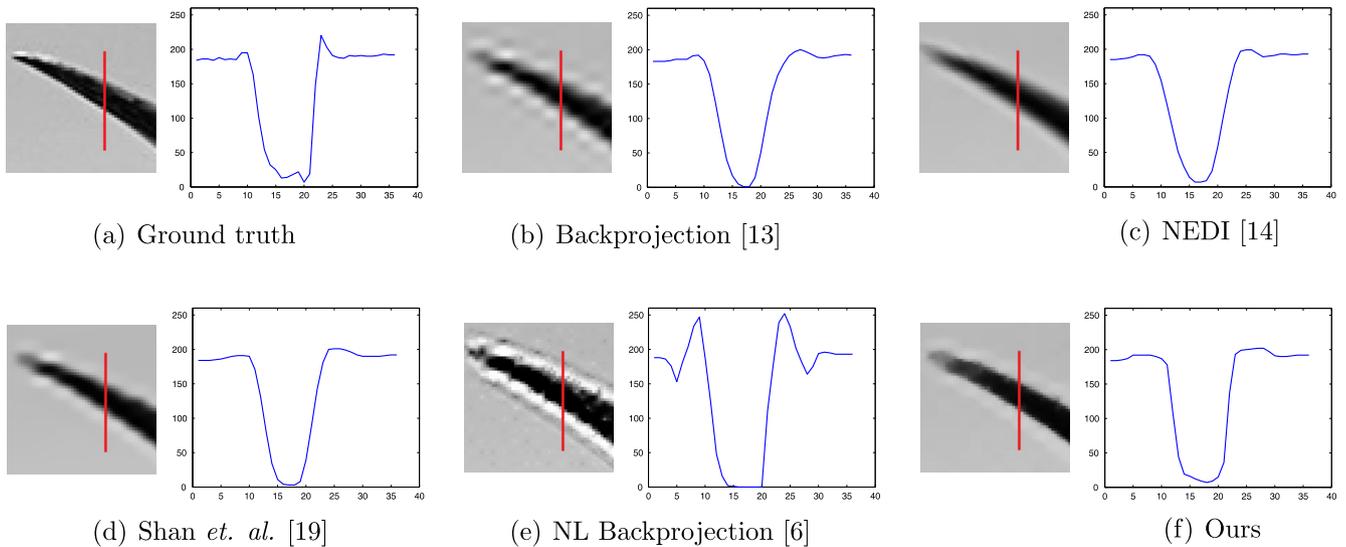


Fig. 21. Penguin 1-D profiles. We extract 1-D intensity profiles (along the red lines) from the results of the Penguin image of Fig. 20, and plot them. Most other algorithms are not able to produce ramp transitions as sharp as ours. Our profile bears the closest resemblance to the ground truth. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

bear closer resemblance to the ground truth images when compared to them. In some results, we also extract a few 1-D pixel profiles, in order to visually examine the quality of the reconstructed ramps of our method vs the others, and compare to the ground truth.

We also show some results on LR images that do not have ground truth HR versions, but have been used by several recent methods and have been provided online [11,8,7]. This allows us to visually compare our results with these methods, albeit without any ground truth.

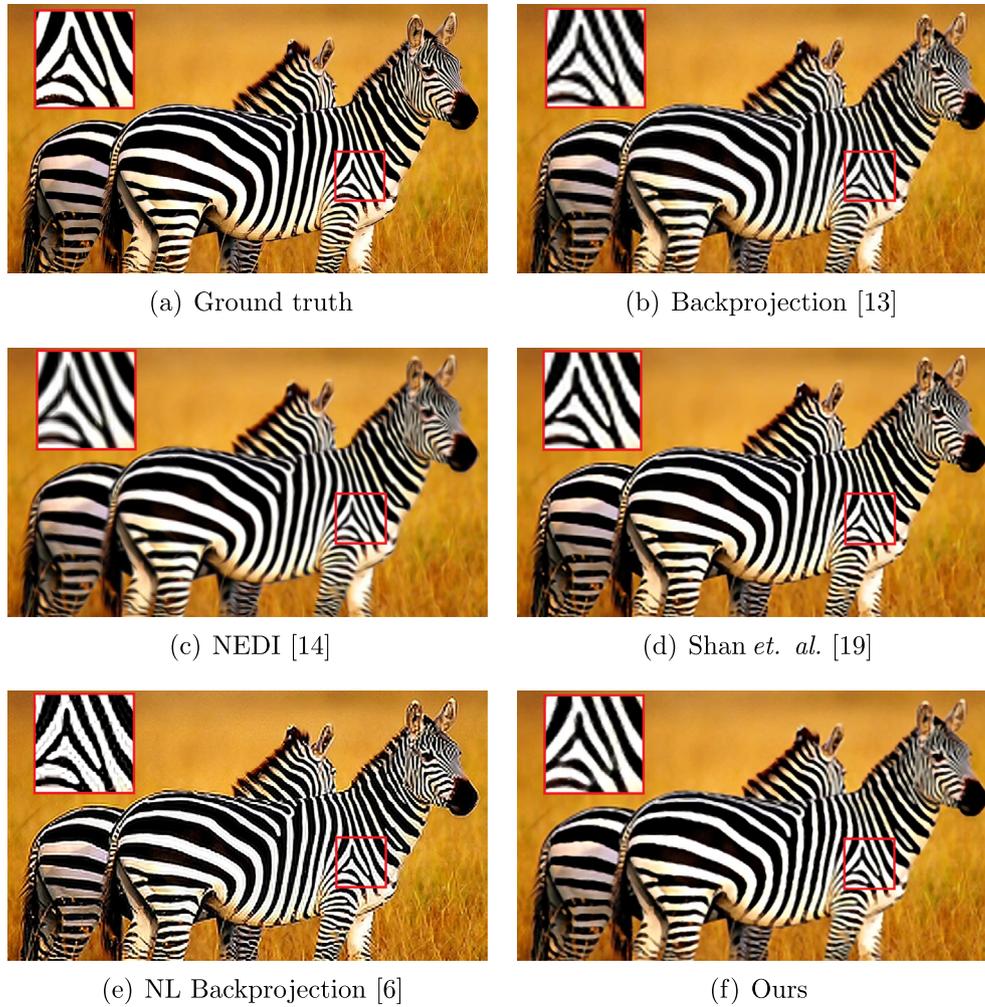


Fig. 22. Zebra image (4X). The stripes in our result are sharp, without any ringing effect such as in Shan et al. [19], and match the original image closely. NL backprojection [6] maintains sharpness but introduces spurious black artifacts in the interior of the white stripes. Best viewed when zoomed in.

Table 2
Comparison of PSNR (db) and structural similarity measure (SSIM) [31] for the Penguin, Zebra, and Flower images. Values in bold indicate the best performance among the different methods.

Image	Backproj. [13]	NEDI [14]	Shan [19]	NL Backproj. [6]	Ours
Penguin	26.15 .8576	24.32 .8565	26.15 .8766	23.54 .8170	25.46 .8799
Zebra	20.06 .6755	18.74 .6539	20.20 .7123	19.68 .6750	20.23 .7145
Flower	29.61 .8574	27.32 .8185	29.99 .8554	26.03 .7734	30.08 .8591

8.3. Comparison with learning based methods

We first compare our results to the gradient profile prior (GPP) [21,20] method, which is a learning based approach, most related to our method. We also compare to the patch based dictionary learning approach of Yang et al. [28]. In addition, as baselines we also compare to the classical bicubic interpolation as well as the backprojection algorithm [13], which is equivalent to setting $\lambda = 0$ in (8).

Fig. 14 shows our comparisons on the Leaves image. Both the GPP based approach [21] as well as the dictionary based approach [28] do not produce sufficiently sharp results. Our edges are reconstructed better and are most similar to the ground truth image shown in Fig. 14(a). In Fig. 15, we show a 1-D profile extracted

from the Leaves image. We can see that the ramp transitions are better reconstructed by our algorithm as compared to the others. Thin/fine structures are blurred and flattened out by other methods, but our algorithm is able to better preserve such structures.

Fig. 16 compares results on the Bird image. Here we highlight an important drawback of gradient based edge priors such as GPP [21]: As shown in Fig. 16(d), the GPP result fails to accurately reproduce the color of the sky, as compared to the ground truth figure of Fig. 16(a). The gradient domain constraint in [21] does not enforce brightness consistency. Edges are also not as sharp as those obtained by our method in Fig. 16(f). Although sharper than GPP, the dictionary based method by Yang et al [28] produces ringing artifacts along edges, such as along the bird's beak in Fig. 16(e). Our result is free of such artifacts.

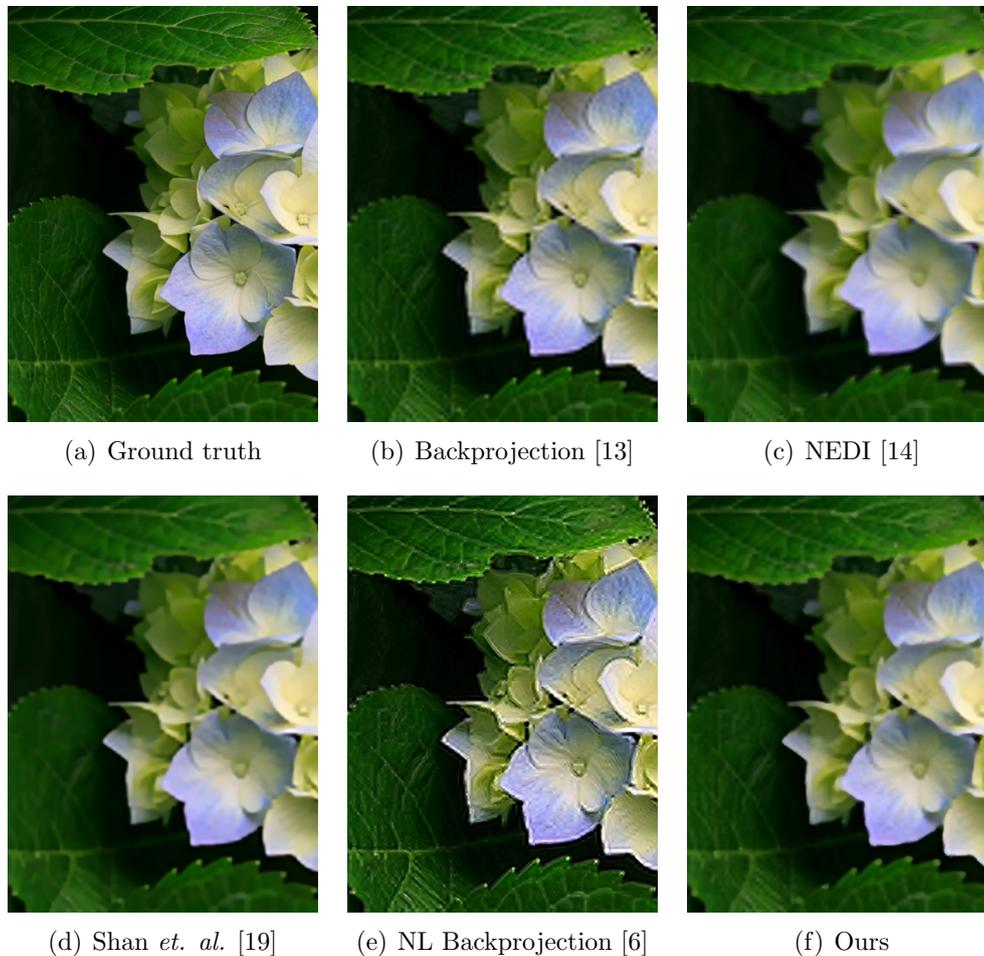


Fig. 23. Flower image (4X). NL backprojection [6] maintains sharpness but introduces artifacts near edges. Shan et al. [19] yields results comparable to ours, but fine textural details (such as in the interior of leaves) appear slightly faded.

Fig. 18 and Fig. 17 show two more similar comparisons. Our results are consistently sharper than other methods, and bear closer visual resemblance to the ground truth, with almost no ringing artifacts. Fig. 19 shows 1-D profiles extracted from the results of Fig. 18. Our ramps bear the closest resemblance to the ground truth. Other methods produce ramp transitions that are less sharp.

For these images, we tabulate the PSNR (db) and SSIM [31] values in Table 1. Our numerical results are significantly better than GPP [21] and also the bicubic and backprojection [13] baselines. Our values are quite close to those of Yang et al. [28], although visually our results appear better, particularly along strong edges.

We also compare our results with the edge statistics driven method of Fattal [7] in Figs. 24 and 25. Note that no ground truths are available for these images. In the *Wheel* image of Fig. 24, the result of Fattal [7] exhibits more ringing artifacts than our result, and it also appears overly smooth in places. In the *Sculpture* image of Fig. 25, Fattal [7] exhibits disparity in brightness due to reasons mentioned earlier. Our method, while also being edge driven, is fundamentally free from this drawback.

8.4. Comparisons with interpolation and reconstruction based methods

We now compare our results to some methods that do not utilize any learning based prior knowledge. In particular, we compare our results to those of Shan et al. [19] that is based on iterative feedback based filtering. We also compare to the edge directed interpolation method called NEDI [14], and non-local (NL)

backprojection [6] which is an edge-aware extension to the classical backprojection algorithm [13].

Fig. 20 shows our comparison on the *Penguin* image. NL backprojection [6] overcomes some of the excessive ringing effect seen in the classical backprojection algorithm [13], but also introduces other artifacts along the edges. The edge directed interpolation method in NEDI [14] produces overly smooth results. Shan et al. [19] produces good results, but shows slightly more ringing artifacts as compared to our result, such as along the beak of the penguin. We again show 1-D profiles in Fig. 21, extracted from the *Penguin* results. Our algorithm is able to accurately reconstruct the sharpness of the ground truth ramps. Other methods typically are not that sharp.

In the *Zebra* image of Fig. 22, Shan et al. [19] exhibits ringing or zig-zag like effects along the stripe edges. NL backprojection [6], although sharp, creates unwanted dark streaks in the interior of the white stripes. This is better visualized by zooming in on Fig. 20(e). Our result is free of such artifacts.

Fig. 23 shows another set of results. Although the ringing effect of Shan et al. [19] is not very evident in this result, fine textural details (such as in the interior of leaves) seem to be slightly washed out as compared to our result.

We show quantitative results of these three images in Table 2. Overall, our results are better quantitatively as well.

8.5. Comparison with self similarity based methods

Freedman [8] and Glasner [11] have proposed SR methods that exploit self similarity within images. While they produce visually

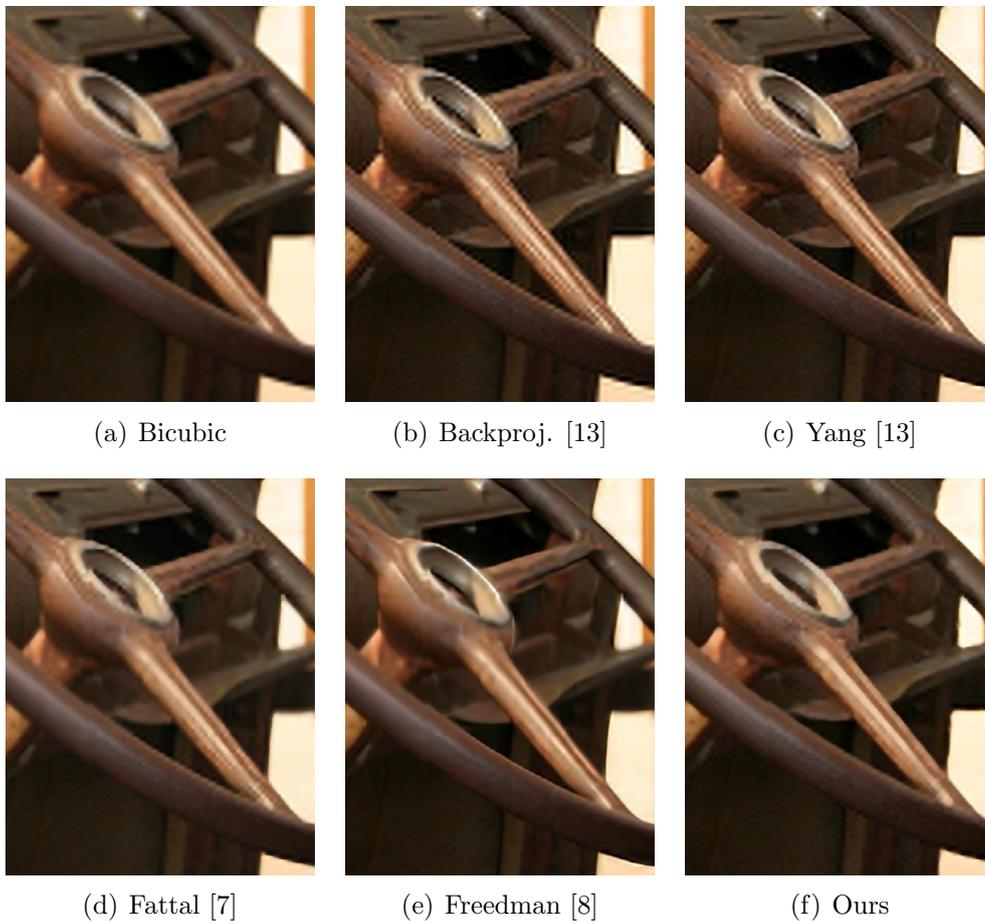


Fig. 24. *Steering Wheel* image (4X). Our result shows less ringing than Fattal [7] and Yang [28]. The self-similarity approach of Freedman [8] does not look very photo-realistic (e.g. the circular rim in the center of the wheel), and some textural details are smoothed out. Our result seems well balanced overall. No ground truth is available for this image.

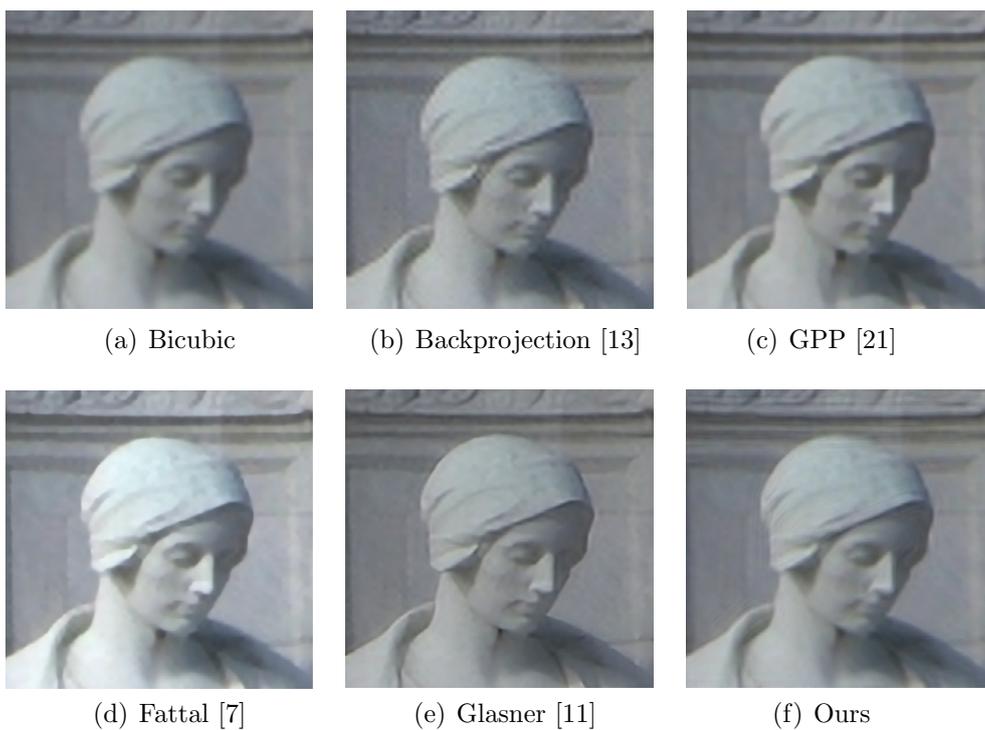


Fig. 25. *Sculpture* (4X). Fattal [7] does not maintain the original brightness level of the image. Our result for this image looks better than the edge based methods of GPP [21] and Fattal [7]. Glasner et al. [11] appears slightly sharper than ours. No ground truth is available for this image.

pleasing results with sharp edges, they often do not appear very photorealistic, and tend to smooth out fine details.

Fig. 24(e) shows the result of the method by Freedman [8] on the *Wheel* image. The circular rim in the center of the steering wheel appears un-natural as compared to the other results, and fine textural details also appear to be smoothed out as compared to our result in Fig. 24(f). Fig. 25(e) shows the result of Glasner et al. [11] on the *Sculpture* image. Objective evaluation on this image is difficult since the ground truth HR image is not available.

9. Discussion and conclusion

Quantifying structure accurately is a fundamental problem in several low level vision tasks. Conventional methods are based on using pre-defined filters and linear convolution. In this paper, we have shown some drawbacks of such a formulation, for the single image SR problem. We have presented a ramp profile based model of structure around image edges for learning a prior for SR, that overcomes the important drawbacks of edge based priors using gradients. In addition, we have also proposed in our model a simple, but novel idea of incorporating dependency between closely spaced edges, while recovering the HR image. Our method is based on characterizing structure around edges, as detected by a low level segmentation procedure. For dealing with region interiors, our algorithm simply defaults to the classical backprojection algorithm. We have obtained better results than several state-of-the-art techniques. Our primary improvement and contribution lies in the better reconstruction of edges as compared to other methods.

A limitation of our algorithm is the lack of robustness to noise. Our definition of ramps as described in the paper assumes a relatively noise free image. In the presence of noise, spurious or distorted ramps may be detected which may hinder performance in the subsequent steps, particularly during the learning phase. In such noisy scenarios, we preprocess the images with a denoising algorithm [4,3] before using the proposed SR method.

Note that noise sensitivity is an issue for other edge based SR methods as well [21,20,7]. Presence of noise significantly affects gradient computations on which algorithms such as GPP [21] are based. Patch based methods generally tend to perform relatively better in presence of noise [28].

In our algorithm, the relatively smooth region interiors are super-resolved using the backprojection constraint alone. Although from a perceptual standpoint, we are justified in focusing on edges and the structures around them for the SR problem, improvement in results may be expected through better modeling of region interiors as well. One approach of doing so could be the integration of both edge based and patch based priors, in order to reap the advantages of each. This would also lend some noise

robustness to our approach. We would be exploring these ideas in the future.

References

- [1] N. Ahuja, A transform for multiscale image segmentation by integrated edge and region detection, IEEE TPAMI (1996).
- [2] E. Akbas, N. Ahuja, From ramp discontinuities to segmentation tree, in: ACCV, 2009.
- [3] A. Buades, B. Coll, J.M. Morel, A non-local algorithm for image denoising, in: CVPR, 2005.
- [4] K. Dabov, A. Foi, V. Katkovnik, K. Egiazarian, Image denoising by sparse 3-d transform-domain collaborative filtering, IEEE Trans. Image Proc. (2007).
- [5] S. Dai, M. Han, W. Xu, Y. Wu, Y. Gong, Soft edge smoothness prior for alpha channel super resolution, in: CVPR, 2007.
- [6] W. Dong, L. Zhang, G. Shi, X. Wu, Nonlocal back-projection for adaptive image enlargement, in: ICIP, 2009.
- [7] R. Fattal, Image upsampling via imposed edge statistics, ACM Trans. Graph. (2007).
- [8] G. Freedman, R. Fattal, Image and video upscaling from local self-examples, ACM Trans. Graph. (2010).
- [9] W. Freeman, E. Pasztor, Learning low-level vision, in: ICCV, 1999.
- [10] W.T. Freeman, E.C. Pasztor, Learning low-level vision, IJCV (2000).
- [11] D. Glasner, S. Bagon, M. Irani, Super-resolution from a single image, in: ICCV, 2009.
- [12] J. Huang, D. Mumford, Statistics of natural images and models, in: CVPR, 1999.
- [13] M. Irani, S. Peleg, Improving resolution by image registration, CVGIP (1991).
- [14] X. Li, M. Orchard, New edge-directed interpolation, IEEE Trans. Image Proc. (2001).
- [15] S. Mallat, G. Yu, Super-resolution with sparse mixing estimators, IEEE Trans. Image Proc. (2010).
- [16] B. Morse, D. Schwartzwald, Image magnification using level-set reconstruction, in: CVPR, 2001.
- [17] P. Moulin, J. Liu, Analysis of multiresolution image denoising schemes using generalized gaussian and complexity priors, IEEE Trans. Inf. Theory (1999).
- [18] S. Roth, M. Black, Fields of experts: a framework for learning image priors, in: CVPR, 2005.
- [19] Q. Shan, Z. Li, J. Jia, C.-K. Tang, Fast image/video upsampling, ACM Trans. Graph. (2008).
- [20] J. Sun, J. Sun, Z. Xu, H.-Y. Shum, Image super-resolution using gradient profile prior, in: CVPR, 2008.
- [21] J. Sun, J. Sun, Z. Xu, H.-Y. Shum, Gradient profile prior and its applications in image super-resolution and enhancement, IEEE Trans. Image Proc. (2011).
- [22] J. Sun, N.-N. Zheng, H. Tao, H.-Y. Shum, Image hallucination with primal sketch priors, in: CVPR, 2003.
- [23] Y.-W. Tai, S. Liu, M. Brown, S. Lin, Super resolution using edge prior and single image detail synthesis, in: CVPR, 2010.
- [24] Y.-W. Tai, W.-S. Tong, C.-K. Tang, Perceptually-inspired and edge-directed color image super-resolution, in: CVPR, 2006.
- [25] L. Wang, S. Xiang, G. Meng, H.-Y. Wu, C. Pan, Edge-directed single image super-resolution via adaptive gradient magnitude self-interpolation, IEEE Trans. CSVT (2013).
- [26] J. Yang, Z. Wang, Z. Lin, S. Cohen, T. Huang, Coupled dictionary training for image super-resolution, IEEE Trans. Image Proc. (2012).
- [27] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution as sparse representation of raw image patches, in: CVPR, 2008.
- [28] J. Yang, J. Wright, T. Huang, Y. Ma, Image super-resolution via sparse representation, IEEE Trans. Image Proc. (2010).
- [29] D.-Y. Yeung, D.-Y. Yeung, Y. Xiong, Super-resolution through neighbor embedding, in: CVPR, 2004.
- [30] X. Zhang, X. Wu, Image interpolation by adaptive 2-d autoregressive modeling and soft-decision estimation, IEEE Trans. Image Proc. (2008).
- [31] Z. Wang, A. Bovik, H. Sheikh, E. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Proc. (2004).