

# AN ADAPTIVE NONPARAMETRIC APPROACH TO RESTORATION AND INTERPOLATION FOR MEDICAL IMAGING

*Hiroyuki Takeda and Peyman Milanfar*

{htakeda,milanfar}@soe.ucsc.edu  
University of California, Santa Cruz

## ABSTRACT

We present the application of a novel nonparametric approach to restoration and interpolation of medical images. The proposed approach is based on the notion of spatially adaptive filtering where locally computed filters adjust to the underlying estimated geometry of the signal of interest. In particular, the approach allows for high performance denoising, restoration and interpolation of images from a variety of modalities using the same mathematical and computational framework.

*Index Terms*— Kernel regression, denoising, interpolation, tomography

## 1. INTRODUCTION

Classical parametric approaches to signal and image processing rely on a specific model of the signal of interest, and seek to compute the parameters of this model in the presence of noise. Examples of this approach are presented in diverse problems ranging from denoising to upscaling and interpolation. A generative model based upon the estimated parameters is then produced as the best estimate of the underlying signal. In contrast to the parametric methods, nonparametric methods rely on the data itself to dictate the structure of the model, in which case this implicit model is referred to as a regression function. With the relatively recent emergence of machine learning methods, kernel methods have become well-known and used frequently for pattern detection and discrimination problems. Surprisingly, it appears that the corresponding ideas in nonparametric estimation (what we here call kernel regression), are not widely recognized or used in the image restoration literature. The B-Spline approach [1, 2] and the wavelet approach [3, 4] are still popular methods for interpolation and denoising, respectively, in medical imaging.

Existing methods for processing, reconstruction, and interpolation of spatial data, particularly in medical imaging, are often too strongly dependent upon underlying assumptions about signal and noise models. We have developed a more “universally” applicable and robust approach based on adaptive nonparametric statistics (i.e. kernel regression or KR) for processing and reconstruction [5].

## 2. REGULARIZED ADAPTIVE KERNEL REGRESSION

In this section, we briefly review the KR framework, and provide some intuition behind it. After that, including the distortion effect, such as a downsampling operation, we derive a regularized KR method for the medical imaging application.

### 2.1. Classic Kernel Regression

Defining the degraded/transformed function as  $z(\mathbf{x}) = \psi\{u(\mathbf{x})\}$ , we write the data model,

$$y_i = z(\mathbf{x}_i) + \epsilon_i, \quad i = 1, 2, \dots, P, \quad \mathbf{x}_i = [x_{1i}, x_{2i}], \quad (1)$$

where  $y_i$  is a noise-laden sample at  $\mathbf{x}_i$ ,  $\epsilon_i$  is i.i.d zero mean noise, and  $P$  is the total number of samples in a neighborhood (window) of interest, and  $z(\cdot)$  is the (hitherto unspecified) *regression function* to be estimated for now, though  $u(\cdot)$  is eventually the unknown function which we are interested in estimating. The KR framework provides a rich mechanism for computing point-wise estimates of the regression function with minimal assumptions about global signal or noise models.

While the particular form of  $z(\cdot)$  may remain unspecified for now, we can rely on a generic local expansion of the function about a sampling point  $\mathbf{x}_i$ . Specifically, if  $\mathbf{x}$  is near the sample at  $\mathbf{x}_i$ , we have the  $N$ -th order Taylor series,

$$\begin{aligned} z(\mathbf{x}_i) &= z(\mathbf{x}) + \{\nabla z(\mathbf{x})\}^T (\mathbf{x}_i - \mathbf{x}) \\ &\quad + \frac{1}{2} (\mathbf{x}_i - \mathbf{x})^T \{\mathcal{H}z(\mathbf{x})\} (\mathbf{x}_i - \mathbf{x}) + \dots \\ &= \beta_0 + \beta_1^T (\mathbf{x}_i - \mathbf{x}) + \beta_2^T \text{vech}\{(\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T\} + \dots \end{aligned} \quad (2)$$

where  $\nabla$  and  $\mathcal{H}$  are the gradient ( $2 \times 1$ ) and Hessian ( $2 \times 2$ ) operators, respectively, and  $\text{vech}(\cdot)$  is the half-vectorization operator which lexicographically orders the lower triangular portion of a symmetric matrix into a column-stacked vector. Furthermore,  $\beta_0$  is  $z(\mathbf{x})$ , which is the pixel value of interest, and the vectors  $\beta_1$  and  $\beta_2$  contain the first and second order derivatives at  $\mathbf{x}$ , respectively.

Since this approach is based on a *local* representation of say order  $N$ , a logical step to take is to estimate the parameters  $\{\beta_n\}_{n=0}^N$  from all the samples  $\{y_i\}_{i=1}^P$ , while giving the nearby samples higher weights than samples farther away.

A formulation of the fitting problem capturing this idea is to solve the following optimization problem:

$$\min_{\{\beta_n\}_{n=0}^N} \sum_{i=1}^P \left\{ y_i - \beta_0 - \beta_1^T (\mathbf{x}_i - \mathbf{x}) - \beta_2^T \text{vech} \left\{ (\mathbf{x}_i - \mathbf{x})(\mathbf{x}_i - \mathbf{x})^T \right\} - \dots \right\}^2 K_{\mathbf{H}_i}(\mathbf{x}_i - \mathbf{x}) \quad (3)$$

where  $N$  is the regression order and  $K(\cdot)$  is the kernel function (a radially symmetric function, such as Gaussian), i.e.,

$$K_{\mathbf{H}_i}(\mathbf{x}_i - \mathbf{x}) = \frac{1}{2\pi\sqrt{\det(\mathbf{H}_i^T \mathbf{H}_i)}} \exp \left\{ -\frac{(\mathbf{x}_i - \mathbf{x})^T (\mathbf{H}_i^T \mathbf{H}_i)^{-1} (\mathbf{x}_i - \mathbf{x})}{2} \right\}, \quad (4)$$

and  $\mathbf{H}_i$  is the smoothing ( $2 \times 2$ ) matrix which dictates the ‘‘footprint’’ of the kernel function. The simplest choice is  $\mathbf{H}_i = h\mathbf{I}$ , where  $h$  is called the global smoothing parameter. It is naturally simpler to choose a fixed  $\mathbf{H}$  matrix for all the pixels being processed. However, as we describe in Section 2.2, significant advantages are realized by selecting locally adaptive weights.

To sum up so far, the optimization problem (3), which we term ‘‘classic’’ KR eventually provides a point-wise estimator of the regression function (q.v. [5] for derivations). Regardless of the choice of  $K(\cdot)$  and  $N$ , the estimator always yields a weighted linear combination of nearby samples, that is

$$\hat{z}(\mathbf{x}) = \hat{\beta}_0 = \sum_{i=1}^P W_i(K, \mathbf{H}_i, N, \mathbf{x}_i - \mathbf{x}) y_i, \quad \sum_{i=1}^P W_i(\cdot) = 1, \quad (5)$$

where we call  $W_i$  the *equivalent kernel* function for  $y_i$ .

## 2.2. Data-Dependant Weights

One fundamental improvement on the above method can be realized by noting that the KR estimates using (4), independent of the regression order ( $N$ ), are always local *linear* combination on the neighboring samples, i.e., (5) with a fixed kernel function. Hence, they suffer from an inherent limitation due to this local linear action on the data. In [5], we introduced *data-adaptive kernel functions* which rely on not only the spatial distances ( $\mathbf{v} = \mathbf{x}_i - \mathbf{x}$ ), but also on the radiometric properties of nearby samples, i.e.,

$$K_{\mathbf{H}_i}(\mathbf{x}_i - \mathbf{x}) \Rightarrow K_{\text{adapt}}(\mathbf{x}_i - \mathbf{x}, y_i - y). \quad (6)$$

In this paper, we use the concept of steering kernels as introduced in [5]. Briefly speaking, the steering matrix is defined as

$$\mathbf{H}_i^{\text{steer}} = h\mathbf{C}_i^{-\frac{1}{2}}, \quad (7)$$

where  $\mathbf{C}_i$  is a (symmetric) covariance matrix of local gradient vectors. It is well known that the local edge structure is related to the gradient covariance (or equivalent, the locally dominant orientation). Using the steering matrix, the weight



**Fig. 1.** Visual representations of steering kernel weights at different structures.

matrix takes local image structures into account and that enables our estimation to preserve edges and reduce noise in flat regions.

More specifically, a naive estimate of this covariance matrix may be obtained as follows:

$$\hat{\mathbf{C}}_i = \mathbf{J}^T \mathbf{J} \quad (8)$$

with

$$\mathbf{J} = \begin{bmatrix} \vdots & \vdots \\ z_{x_1}(\mathbf{x}_j) & z_{x_2}(\mathbf{x}_j) \\ \vdots & \vdots \end{bmatrix}, \quad \mathbf{x}_j \in \xi_i \quad (9)$$

where  $z_{x_1}(\cdot)$  and  $z_{x_2}(\cdot)$  are the first derivatives along the  $x_1$ - and  $x_2$ -directions, respectively,  $\xi_i$  is a local analysis window around the position of interest  $\mathbf{x}_i$ . The dominant local orientation of the gradients is then related to the eigenvectors of this estimated matrix. The gradients depend on the pixel values, and since the choice of the localized kernels in turn depends on these gradients, it, therefore, follows that the ‘‘equivalent’’ kernels for the data-adapted methods form a locally *nonlinear* combination of the local data. Fig. 1 illustrates steering kernel weights at different structures such as edge, flat, and corner areas. As shown in the figure, steering kernels (weights) capture local image structures. Moreover, steering KR is able to interpolate pixels along a local orientation at an edge area.

While this approach (which is essentially a principal component method to analyze image (orientation) structure [6, 7]) is simple and has nice tolerance to noise, the resulting estimate of the covariance may in general be rank deficient or unstable. Therefore, in this case, care must be taken not to take the inverse of the estimate directly. More details about the estimation of the steering weights and its implementation are available in [5, 8].

## 2.3. Regularized Kernel Regression

Considering the effect of distortion or transformation, instead of  $z$ , the function  $u$  is the one we wish to estimate. Therefore, in place of representing the regression function  $z$  by a local approximation (Taylor series), we apply the KR framework to  $u$ . Due to the operation  $\psi$  in (1), neighboring pixels might

be coupled and it is hard to estimate pixel values of  $u(\mathbf{x})$  individually. Hence, we write the data model in matrix form as

$$\underline{\mathbf{Y}} = \underline{\mathbf{Z}} + \underline{\boldsymbol{\epsilon}} = \underline{\boldsymbol{\Psi}}\underline{\mathbf{U}} + \underline{\boldsymbol{\epsilon}} \quad (10)$$

where  $\underline{\mathbf{Y}}$  is the measured signal,  $\underline{\mathbf{Z}}$  was the regression image to be estimated, but now  $\underline{\mathbf{U}} \in \mathcal{R}^{NM \times 1}$  is the unknown image of interest, and  $\underline{\boldsymbol{\epsilon}}$  and  $\underline{\boldsymbol{\Psi}}$  represents noise and a distortion operation such as downsampling and blurring, respectively. An underlined vector corresponds to a matrix that is lexicographically ordered into a column stack vector:  $\underline{\mathbf{U}} = [u(\mathbf{x}_1), u(\mathbf{x}_2), \dots]^T \in \mathcal{R}^{NM \times 1}$  where  $u(\cdot)$  is the unknown function of interest and  $\mathbf{x}_i$  is a  $2 \times 1$  vector which indicates the pixel coordinate.

While any specific knowledge about the unknown function  $u(\cdot)$  is also unavailable, we again assume that  $u(\cdot)$  is locally differentiable. That is to say, we can express the relationship between two neighboring pixel values,  $u(\mathbf{x}_i)$  and  $u(\mathbf{x}_i + \mathbf{v})$  where  $\mathbf{v} = [v_1, v_2]^T$ , as

$$\begin{aligned} u(\mathbf{x}_i + \mathbf{v}) &= u(\mathbf{x}_i) + u_{x_1}(\mathbf{x}_i) \cdot v_1 + u_{x_2}(\mathbf{x}_i) \cdot v_2 \\ &\quad + u_{x_1^2}(\mathbf{x}_i) \cdot \frac{v_1^2}{2} + u_{x_1 x_2}(\mathbf{x}_i) \cdot v_1 v_2 \\ &\quad + u_{x_2^2}(\mathbf{x}_i) \cdot \frac{v_2^2}{2} + \dots \end{aligned} \quad (11)$$

where  $u_{x_1}(\cdot)$  and  $u_{x_2}(\cdot)$  are the first and second order derivatives along the vertical ( $x_1$ ) direction of  $u$ . Since the expression above is valid for all the pixel coordinates in the unknown image  $\underline{\mathbf{U}}$ , we stack all the local representations into vector form:

$$\begin{aligned} \mathbf{S}_{x_1}^{v_1} \mathbf{S}_{x_2}^{v_2} \underline{\mathbf{U}} &= \underline{\mathbf{U}} + \underline{\mathbf{U}}_{x_1} v_1 + \underline{\mathbf{U}}_{x_2} v_2 \\ &\quad + \underline{\mathbf{U}}_{x_1^2} \frac{v_1^2}{2} + \underline{\mathbf{U}}_{x_1 x_2} v_1 v_2 + \underline{\mathbf{U}}_{x_2^2} \frac{v_2^2}{2} + \dots, \end{aligned} \quad (12)$$

where  $\mathbf{S}_{x_1}^{v_1}$  and  $\mathbf{S}_{x_2}^{v_2}$  are shift operators that shift the image  $\underline{\mathbf{U}}$  along  $x_1$ - and  $x_2$ -directions by the distances  $v_1$  and  $v_2$ , respectively. Considering a local approximation of order  $N$ , we write the above expression as

$$\underline{\mathbf{U}} \approx \mathbf{S}_{x_1}^{-v_1} \mathbf{S}_{x_2}^{-v_2} \mathbb{I}_N \underline{\mathbf{U}}_N \quad (13)$$

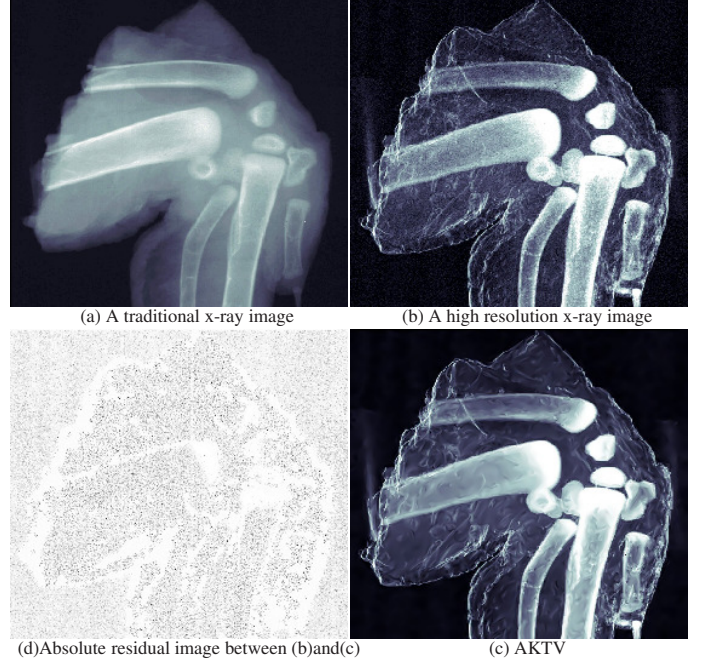
where, for example, when  $N = 2$ , we have

$$\begin{aligned} \mathbb{I}_2 &= \left[ \mathbf{I}, \mathbf{I}_{v_1}, \mathbf{I}_{v_2}, \mathbf{I}_{v_1^2}, \mathbf{I}_{v_1 v_2}, \mathbf{I}_{v_2^2} \right] \\ \underline{\mathbf{U}} &= \left[ \underline{\mathbf{U}}^T, \underline{\mathbf{U}}_{x_1}^T, \underline{\mathbf{U}}_{x_2}^T, \underline{\mathbf{U}}_{x_1^2}^T, \underline{\mathbf{U}}_{x_1 x_2}^T, \underline{\mathbf{U}}_{x_2^2}^T \right]^T, \end{aligned} \quad (14)$$

and  $\mathbf{I}_{v_1} = \text{diag}\{v_1, \dots, v_1\}$ . The regression order ( $N$ ) controls the bias/variance tradeoff of the estimate: the larger  $N$ , the smaller the bias and the larger the variance become. However, in terms of the computational efficiency,  $N = 0, 1$ , and  $2$  are the most reasonable choices for most cases.

Having introduced the local shift approximation (13), the data model (10) then becomes

$$\underline{\mathbf{Y}} \approx \underline{\boldsymbol{\Psi}} \mathbf{S}_{x_1}^{-v_1} \mathbf{S}_{x_2}^{-v_2} \mathbb{I}_N \underline{\mathbf{U}}_N + \underline{\boldsymbol{\epsilon}}. \quad (15)$$



**Fig. 2.** A denoising example: (a) a traditional x-ray image of a chicken wing, (b) a high resolution x-ray image, (c) the denoised image by AKTV, and (d) absolute residual image.

Now, Eqs.(15) and (13) suggest a data fidelity (or likelihood term) and a general (smoothness) prior (or regularization term). That is, the overall cost function to be minimized is proposed as

$$\begin{aligned} C(\underline{\mathbf{U}}_N) &= \sum_{v_1} \sum_{v_2} \left\{ \left\| \underline{\mathbf{Y}} - \underline{\boldsymbol{\Psi}} \mathbf{S}_{x_1}^{-v_1} \mathbf{S}_{x_2}^{-v_2} \mathbb{I}_N \underline{\mathbf{U}}_N \right\|_{\mathbf{W}(\mathbf{v})}^2 \right. \\ &\quad \left. + \eta \left\| \underline{\mathbf{U}} - \mathbf{S}_{x_1}^{-v_1} \mathbf{S}_{x_2}^{-v_2} \mathbb{I}_N \underline{\mathbf{U}}_N \right\|_{\mathbf{W}(\mathbf{v})}^1 \right\}, \end{aligned} \quad (16)$$

where  $\eta$  is the regularization parameter and  $\mathbf{W}(\mathbf{v})$  is a diagonal weight matrix which depends on the shift distance  $\mathbf{v}$ . We call the overall cost function in (16) *Adaptive Kernel Total Variation* (AKTV) [8]. Naturally, with fewer higher order terms in this expansion, the smaller shift distances ( $v_1, v_2$ ) result in a more faithful approximation in (13). Hence, we give larger weights when the shift distance is small. More specifically, the weight matrix is defined as

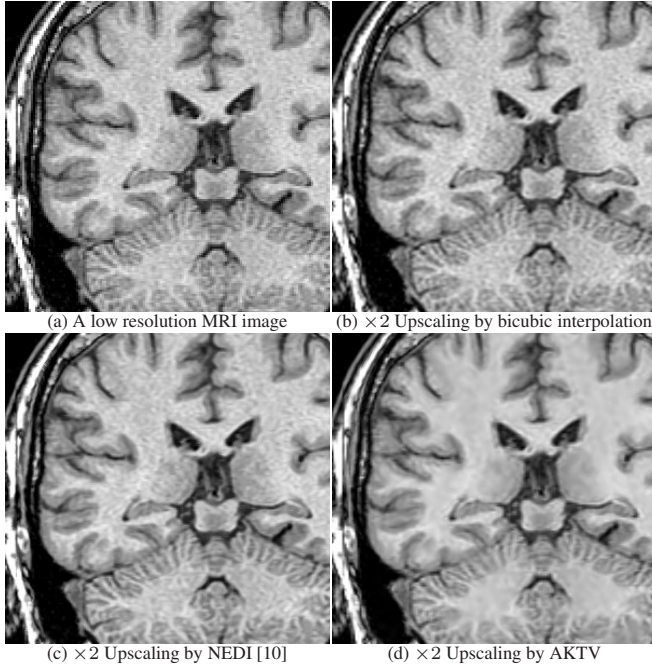
$$\mathbf{W}(\mathbf{v}) = \text{diag}\{K_{\mathbf{H}_1^{\text{steer}}(\mathbf{v})}, K_{\mathbf{H}_2^{\text{steer}}(\mathbf{v})}, \dots\}, \quad (17)$$

where we use steering kernel functions

### 3. EXAMPLES

The first example is image denoising. For this example, the operator  $\underline{\boldsymbol{\Psi}}$  is the identity matrix ( $\underline{\boldsymbol{\Psi}} = \mathbf{I} \in \mathcal{R}^{NM \times NM}$ ), and we used a real high resolution x-ray image of a chicken wing obtained by a new super-resolution x-ray imaging method [9]





**Fig. 3.** An upscaling example: (a) a low resolution MRI image of a human head, and (b)-(d)  $\times 2$  upscaled images by bicubic, NEDI [10], and AKTV, respectively.

shown in Fig. 2(b)<sup>1</sup>. By contrast, Fig. 2(a) shows a traditional X-ray of the same object, which is less noisy, but also considerably less detailed. Therefore, effective denoising of the high-resolution X-ray image can provide unprecedented levels of detail and clarity for this imaging modality. The denoised result by AKTV is shown in Fig. 2(c). Fig. 2(d) illustrates the absolute residual image (the absolute difference between the given noisy image and the denoised image), which is intended to show that AKTV effectively removed noise while preserving structures.

The second example is image upscaling. For upscaling,  $\Psi$  is a downsampling operator:  $\Psi = \mathbf{D} \in \mathcal{R}_{\frac{1}{r}NM \times NM}$ , where  $r$  is the downsampling factor. Fig. 3(a) shows a real low-resolution MRI image ( $256 \times 256$ ) of a human head, which we obtained from the Whole Brain Atlas<sup>2</sup>. The upscaled images with  $r = 2$  by bicubic interpolation, NEDI, and our approach are shown in Figs. 3(b)-(d), respectively. The example shows that the proposed method, AKTV, is capable of upscaling and denoising an image simultaneously, and such a one-step approach is generally more suitable than a multiple-step approach (e.g., denoising followed by upscaling).

#### 4. CONCLUSION AND FUTURE WORK

In this paper, we illustrated the application of an advanced nonparametric approach to the treatment of images from a

<sup>1</sup>The image is available at <http://blogs.zdnet.com/emergingtech/?p=983>

<sup>2</sup><http://www.med.harvard.edu/AANLIB/cases/caseNA/gr/cor/051.png>

variety of modalities. The approach is sufficiently general so that, as illustrated in Section 3, it is applicable a wide variety of problems including denoising, deblurring [8] and interpolation. It is of particular note that the approach makes minimal assumptions about the global structure of the signal and noise, and is therefore quite generally useful. It is also worth stating that the distortion operator,  $\Psi$ , can be, for instance, a combination of Radon transform and downsampling operation. Applications for 3-dimensional processing are also important, and the proposed approach can be easily extended for 3-D data.

#### 5. REFERENCES

- [1] P. Thevenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Transactions on Medical Imaging*, vol. 19, no. 7, pp. 739–758, July 2000.
- [2] T. M. Lehmann, C. Gonner, and K. Spitzer, "B-spline interpolation medical image processing," *IEEE Transactions on Medical Imaging*, vol. 20, no. 7, pp. 660–665, July 2001.
- [3] P. Bao and L. Zhang, "Noise reduction for magnetic resonance images via adaptive multiscale products thresholding," *IEEE Transactions on Medical Imaging*, vol. 22, no. 9, pp. 1089–1099, September 2003.
- [4] A. M. Wink and J. B. T. M. Roerdink, "Denoising functional MR images: A comparison of wavelet denoising and Gaussian smoothing," *IEEE Transactions on Medical Imaging*, vol. 23, no. 3, pp. 374–387, March 2004.
- [5] H. Takeda, S. Farsiu, and P. Milanfar, "Kernel regression for image processing and reconstruction," *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 349–366, February 2007.
- [6] X. Feng and P. Milanfar, "Multiscale principal components analysis for image local orientation estimation," *Proceedings of the 36th Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA*, November 2002.
- [7] R. Mester and M. Muhlich, "Improving motion and orientation estimation using an equilibrated total least squares approach," *Proceedings of IEEE International Conference in Image Processing*, pp. 929–932, 2001.
- [8] H. Takeda, S. Farsiu, and P. Milanfar, "Deblurring using regularized locally-adaptive kernel regression," *IEEE Transactions on Image Processing*, vol. 17, no. 4, pp. 550–563, April 2008.
- [9] P. Thibault, M. Dierolf, A. Menzel, O. Bunk, C. David, and F. Pfeiffer, "High-resolution scanning x-ray diffraction microscopy," *Science*, vol. 321, pp. 379–382, July 2008, Issue 5887.
- [10] X. Li and M. T. Orchard, "New edge-directed interpolation," *IEEE Transactions of Image Processing*, vol. 10, no. 10, pp. 1521–1527, October 2001.