SUPERRESOLUTION OF IMAGES BASED ON LOCAL CORRELATIONS

Frank M. Candocia
candocia@cnel.ufl.edu

Jose C. Principe
principe@cnel.ufl.edu

Computational NeuroEngineering Laboratory
Dept.of Electrical and Computer Engineering
University of Florida, Gainesville, FL. 32611

ABSTRACT

An adaptive two step paradigm for the superresolution of optical images is developed in this paper. The procedure locally projects image samples onto a family of kernels that are learned from image data. First, an unsupervised feature extraction is performed on local neighborhood information from a training image. These features are then used to cluster the neighborhoods into disjoint sets for which an optimal mapping relating homologous neighborhoods across scales can be learned in a supervised manner. A superresolved image is obtained through the convolution of a low-resolution test image with the established family of kernels. Results demonstrate the effectiveness of the approach.

Keywords: superresolution (super-resolution) image processing, magnification, interpolation, linear and nonlinear associative memories, local or interblock correlation, scale interdependence, homologous neighborhoods, modular and adaptive systems.

Appears in the IEEE Transactions on Neural Networks, Vol. 10, No. 2, pp. 372-380, March 1999.

I. INTRODUCTION

Superresolution is the process of obtaining an image at a resolution higher than that afforded by the physical sensor. Superresolution has been used in obtaining high quality image prints and has found applications in areas such as surveillance and automatic target recognition [1,2]. This paper addresses the issue of image magnification (in optical images the process is also referred to as interpolation, zooming, enlargement, etc.) from a finite set of collected data sampled below the Nyquist rate or which had to be antialiased prior to sampling.

Commonly, magnification is accomplished through convolution of the image samples with a single kernel - typically the bilinear, bicubic [3] or cubic B-spline kernel [4]. The mitigation of aliasing by this type of linear filtering is very limited. Magnification techniques based on a priori assumed knowledge are the subject of current research. Directional methods [5,6] examine an image's local edge content and interpolate in the low frequency direction (along the edge) rather than in the high frequency direction (across the edge). Multiple kernel methods typically select between a few ad hoc interpolation kernels [7]. Orthogonal transform methods focus on the use of the discrete cosine transform (DCT) [8,9] and the wavelet transform [10]. Variational methods formulate the interpolation problem as the constrained minimization of a functional [11,12]. An extended literature survey discussing these methods at great length has been provided by Candocia [13]. It should be noted that these techniques make explicit assumptions regarding the character of the analog image.

The superresolution problem is known to be ill-posed [2]. Thus, obtaining a desirable solution requires making reasonable assumptions about the nature of the true image (the most conventional assumption is smoothness). The approach presented herein is novel, and addresses the ill-posed nature of superresolution by assuming that similar (correlated) neighborhoods remain similar across scales, and that this a priori structure can be *learned locally* from available image samples across scales. Such local information extraction has been prominent in image compression schemes for quite some time as evidenced by JPEG [14] and PCA-based [15] approaches which typically compress the set of

nonoverlapping subblocks of an image. Recent compression approaches also exploit the interblock correlation between subblocks [16,17]. The goal is to divide the set of subblocks into a finite number of disjoint sets that can individually be represented more efficiently than was the original set. Our approach is similar in spirit in that we exploit interblock correlation for mapping similar overlapping neighborhoods to their high-resolution counterparts. However, no one proposed the use of this information to create constraints that can be used to superresolve images. We further show that a very simple connectionist architecture can learn this structure effectively. Moreover, our approach is shown to be equivalent to convolution with a family of kernels established from an available image and "tuned" to its local characteristics, which represents an extension to conventional sampling theory concepts.

In section II of this paper we relate single and multiple kernel approaches and formulate the superresolution process as a filtering (convolution) of the image samples. Section III describes the architecture implemented that performs the filtering described in section II. The feature extraction and association of homologous neighborhoods for superresolution is discussed here. Section IV describes the results presented in this paper. Section V discusses issues pertaining to the superresolution approach presented and section VI concludes the paper.

II. KERNEL-BASED APPROACHES

II-*A. SINGLE KERNEL*

An interpolated image can be obtained by expanding the samples of a low-resolution image $x_l[n_1, n_2]$ and convolving with a sampled interpolation kernel [18]. For an expansion rate of $G_1 \times G_2$, where G_1, G_2 are whole numbers greater than 1, our expanded image is given by

$$x_e[n_1, n_2] = \begin{cases} x_l\left[\frac{n_1}{G_1}, \frac{n_2}{G_2}\right] & n_1 = 0, \pm G_1, \pm 2G_1, \cdots \\ & n_2 = 0, \pm G_2, \pm 2G_2, \cdots \\ 0 & \textit{otherwise} \end{cases} \quad (1)$$

and the corresponding interpolation kernel, obtained by sampling a continuous kernel, is denoted $k[n_1, n_2]$. The interpolated image $\hat{x}_h[n_1, n_2]$, which estimates the true image $x_h[n_1, n_2]$, is

$$\hat{x}_h[n_1, n_2] = x_e[n_1, n_2] ** k[n_1, n_2] \tag{2}$$

where $**$ denotes 2D convolution. This form of interpolation is a linear filtering that processes the image similarly throughout, i.e. it uses the same linear combination of image samples in determining interpolated points.

II-*B. FAMILY OF KERNELS*

The kernel family approach is a scheme where the interpolation kernel used depends on the local characteristics of the image [19]. This is formulated as

$$\hat{x}_h[n_1, n_2] = x_e[n_1, n_2] ** k_{c,l}[n_1, n_2] \tag{3}$$

The subscripts *c* and *l*, which are functions of image location, select a kernel based on the local image characteristics about the point of interest. The family of kernels is given by $\{ k_{c,l}[n_1, n_2] : c = 1,...,C; l = 1,...,L \}$. *C* represents the number of established local image characteristics (features) from which to compare local neighborhood information and *L* is the number of kernels created per feature. In summary, eqn. (3) describes a convolution with a shift-varying kernel. It is a generalization of eqn. (2) and defaults to the standard convolution of eqn. (2) when $C,L = 1$.

In the family of kernels approach, two main issues must be addressed: how to design the kernel family and how to select between kernels once they have been established. These issues are now addressed.

II-*C. A PROPOSED METHODOLOGY FOR SUPERRESOLUTION OF IMAGES*

As mentioned earlier, the problem of superresolution is ill-posed. As such, additional information apart from the collected samples is needed to obtain a solution. Rather than assume smoothness or rely on other typical constraints, we choose to assume that a given class of images contains *similar*

information locally and that this similarity holds across scales. So the fundamental problem is to devise a superresolution scheme that will be able to decide similarity of local information and capture similarities across scales in an automated fashion. The family of kernels approach (eqn. (3)) does this naturally. When the image space is divided into regions that cluster similar local information, a countable set of possibilities is created which associates each pixel with a member of the family. For each local region an interpolating kernel $k_{c,l}$ is created to preserve the information across scales. We choose an adaptive scheme to design the kernels $k_{c,l}$ since we can design optimal mappers given a representative set of training images. We further expect that, if the local regions are small enough, the information will generalize across images. When a new image is presented, the kernel that best reconstructs each local region is selected automatically and the reconstruction will appear at the output.

The division of the signal space in local regions will be implemented via a vector quantization (VQ) algorithm as proposed by [20]. Each Voronoi cell resulting from the VQ will be linked to a linear associative memory (LAM) trained to find the best mapping between the low-resolution image and the high-resolution image, hence capturing the information across scales. In other words, the assumption we make is that the information embodied in the codebook vectors and LAMs describes the relation (mapping) between a low-resolution neighborhood and its high-resolution counterpart, hence implementing eqn. (3). As such, our approach does not require the assumptions typically needed to obtain a reasonable solution to the ill-posed superresolution problem.

III. AN ARCHITECTURE FOR KERNEL FAMILY SUPERRESOLUTION

Fig. (1) illustrates the proposed architecture for superresolving images using a family of kernels. As we proceed, the relation between the architecture and eqn. (3) will be elucidated. The purpose of data clustering is to partition the low-resolution image neighborhoods into a finite number of clusters where the neighborhoods within each cluster are similar in some sense. Once the clusters are established, a set of kernels can be developed which optimally transform each clustered neighborhood into its corresponding high-resolution neighborhood. The subsections that follow discuss how the kernel family,

implemented here as linear associative memories (LAMs) in fig. (1), is established and then used for optical image superresolution.

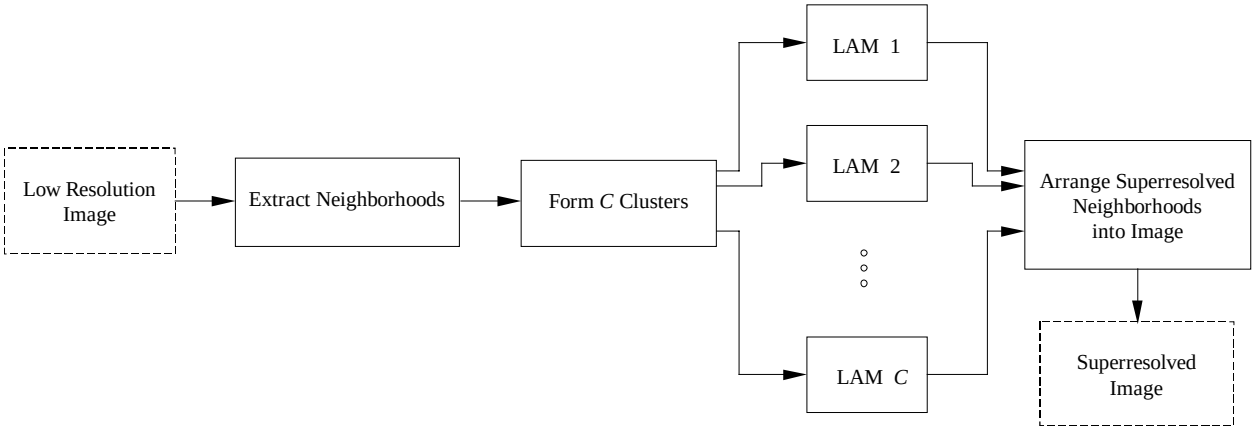


Figure 1. Superresolution architecture based on local correlations. This paradigm performs the equivalent operation of convolution with a family of kernels.

III-*A. THE TRAINING DATA*

Ideally, the low and high-resolution data sets used to train the LAMs of fig. (1) would each encompass the same scene and have been physically obtained by hardware with different but known resolution settings. Such data collection is not common. Instead, the low-resolution counterparts of the given images are simulated via decimation – which is represented by the $\downarrow G_1 \times G_2$ block. Here, the simulated low-resolution images will be obtained following the image acquisition model discussed in [12] which is given by

$$x_l[n_1, n_2] = \frac{1}{G_1 G_2} \sum_{m_1 = G_1 n_1}^{G_1(n_1+1)-1} \sum_{m_2 = G_2 n_2}^{G_2(n_2+1)-1} x_h[m_1, m_2]. \tag{4}$$

Notice that $m_i = 0,1,...,M_i - 1$ and $n_i = 0,1,...,N_i - 1$ such that $M_i = G_i N_i$ for $i = 1,2$. From eqn. (4), it is apparent that the decimation model used employs the averaging of nonoverlapping image neighborhoods as discussed in [12]. After the low-resolution image is synthesized the training of the superresolution architecture proceeds as depicted in fig. (2).

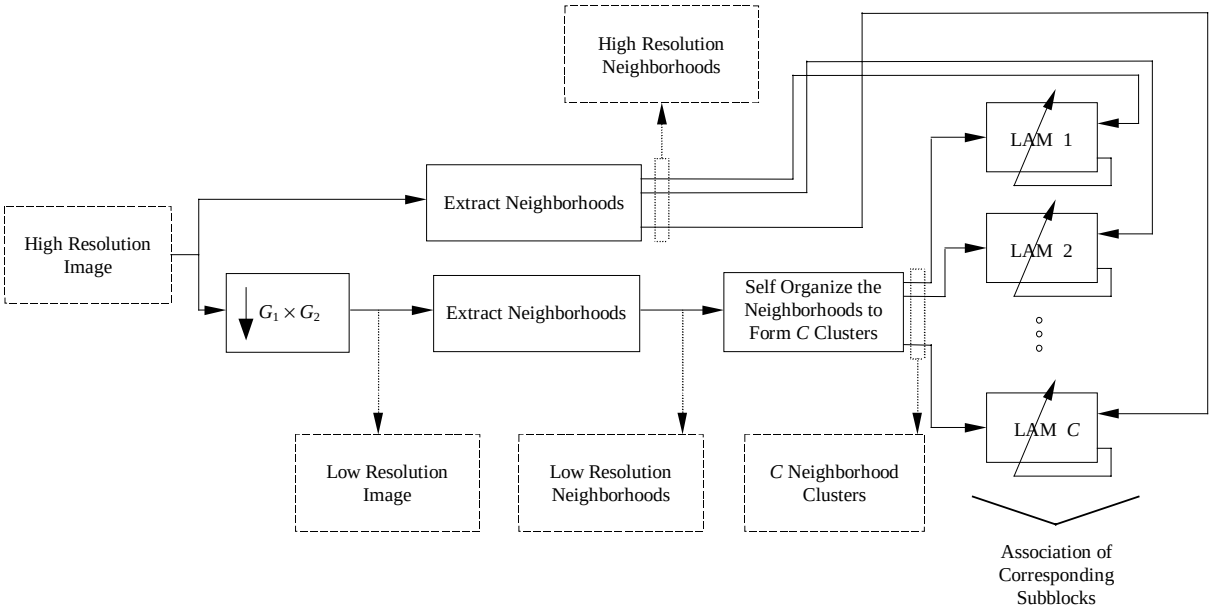


Figure 2. Training approach for the superresolution architecture.

III-*B. CLUSTERING OF DATA*

The neighborhoods considered consist of all the *overlapping* $H_1 \times H_2$ neighborhoods of the low-resolution image $x_l[n_1, n_2]$. The set of these $N = (N_1 - H_1 + 1)(N_2 - H_2 + 1)$ neighborhoods in the low-resolution image are given by

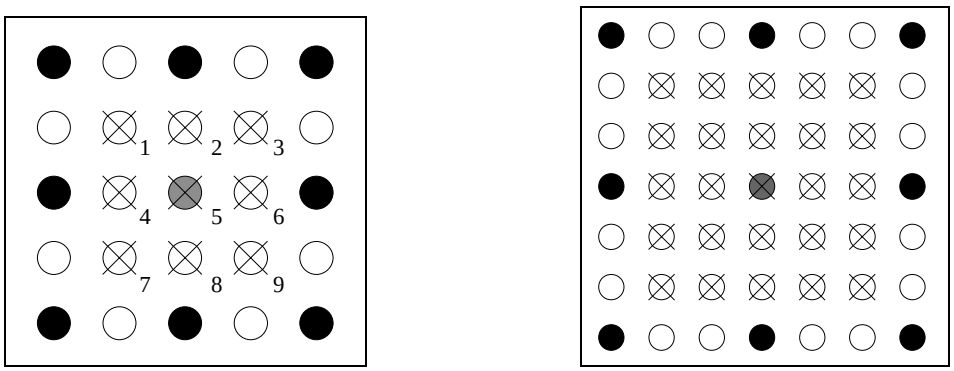
$$X = \left\{ x_l[m_1 : m_1 + H_1 - 1, m_2 : m_2 + H_2 - 1] \right\}\Big|_{m_1=0,\ldots,N_1-H_1,\, m_2=0,\ldots,N_2-H_2} \tag{5}$$

and can be represented by the matrix $\mathbf{X} \in \Re^{H_1 H_2 \times N}$ whose columns are the set of vectors $\{\mathbf{x}_r\}_{r=1}^{N}$ where $\mathbf{x}_r$ is a "vectorized" 2D neighborhood. Each low-resolution neighborhood is paired with its $(2G_1 - 1) \times (2G_2 - 1)$ homologous high-resolution neighborhood. Specifically, these high-resolution neighborhoods are described by

$$S = \left\{ \begin{matrix} x_h[G_1 m_1 + \phi_1 + 1 : G_1(m_1 + 2) + \phi_1 - 1, \\ G_2 m_2 + \phi_2 + 1 : G_2(m_2 + 2) + \phi_2 - 1] \end{matrix} \right\}\Bigg|_{m_1=0,\ldots,N_1-H_1,\, m_2=0,\ldots,N_2-H_2} \tag{6}$$

where $\phi_i = \frac{G_i(H_i - 3)}{2}$ and $(i = 1,2)$. The neighborhoods in S can be represented by a matrix $\mathbf{S} \in \Re^{(2G_1-1)(2G_2-1) \times N}$ similarly to the representation used in $\mathbf{X}$. These low and high-resolution

neighborhoods are depicted in fig. (3) where the shaded circles represent a low-resolution neighborhood. For the case of $G_1 = G_2 = 2$ in fig. (3a), the shaded circles are used to construct the crossed circles about the center of the low-resolution neighborhood. Note that if we elect not to construct the center pixel, we are interpolating locally about the observed image samples. If we elect to construct the center pixel (along with the other crossed circles), we are allowing for the ability to change a "noisy" observed sample. Fig. (3b) similarly illustrates this for the case of $G_1 = G_2 = 3$.



(a) (b)

Figure 3. Local image neighborhoods and the pixels they superresolve. Each circle represents a 2D high-resolution image pixel. The shaded circles are the low-resolution image pixels obtained via decimation of the high-resolution image. The gray pixel is the center of the low-resolution neighborhood. Each $H_1 \times H_2$ low-resolution neighborhood constructs a $(2G_1 - 1) \times (2G_2 - 1)$ high-resolution neighborhood about the low-resolution neighborhood's center - these are depicted by the crossed circles. The numbers are a convention used to distinguish between constructed pixels in this neighborhood. (a) Decimation factor $G_1 = G_2 = 2$. (b) Decimation factor $G_1 = G_2 = 3$.

In establishing our family of kernels, we have chosen to associate the *structure* between the neighborhoods in **X** and **S,** *not* the observed samples themselves. The structure of a neighborhood is defined as the neighborhood with its mean subtracted out; each neighborhood thus becomes a vector whose component mean is zero. This kind of preprocessing allows us to categorize neighborhoods sharing a particular characteristic, i.e. they could be smooth, edgy at a particular orientation, etc., as belonging to the same class regardless of the average intensity of the neighborhood. The structure $\mathbf{p}_r$ of

neighborhood $\mathbf{x}_r$ is obtained through multiplication with the square matrix $\mathbf{Z} \in \Re^{H_1H_2 \times H_1H_2}$, i.e. $\mathbf{p}_r = \mathbf{Z}\mathbf{x}_r$ for a single neighborhood or $\mathbf{P} = \mathbf{ZX}$ for all the input neighborhoods where

$$\mathbf{Z} = \frac{1}{H_1H_2}\begin{pmatrix} H_1H_2 - 1 & -1 & \cdots & -1 \\ -1 & H_1H_2 - 1 & & \vdots \\ \vdots & & \ddots & -1 \\ -1 & \cdots & -1 & H_1H_2 - 1 \end{pmatrix}. \tag{7}$$

The desired exemplars associated with **P** are contained in matrix **D**. Each column in **D** is obtained by subtracting the mean of $\mathbf{x}_r$ from its corresponding neighborhood $\mathbf{s}_r$ in **S**. This is done to compensate for the low-resolution neighborhood mean which has been subtracted from $\mathbf{x}_r$ and must be added back after the high-resolution neighborhood structure is created. Specifically, $\mathbf{D} = \mathbf{S} - \mathbf{AX}$ where $\mathbf{A} \in \Re^{(2G_1-1)(2G_2-1)\times H_1H_2}$ is a constant matrix with elements $\frac{1}{H_1H_2}$.

The clusters are formed by performing a vector quantization (VQ) on the space of structural neighborhoods in **P**. This clustering is based on the interblock correlation amongst the neighborhoods in **P** [13]. The VQ was accomplished using Kohonen's self-organizing map [20] for reasons discussed in section V. The VQ operation results in a set of *C* feature vectors $\{\mathbf{f}_c\}_{c=1}^{C}$ where usually $C << N$. The *C* clusters K_c, c = 1,2,…,*C*, formed by our neighborhood and feature vectors are given by

$$K_c = \left\{ \mathbf{p}_r : \| \mathbf{p}_r - \mathbf{f}_c \|_2 < \| \mathbf{p}_r - \mathbf{f}_b \|_2 ; b = 1,2,...,C; b \neq c; r = 1,2,...,N \right\} \tag{8}$$

III-*C. NEIGHBORHOOD ASSOCIATION*

The input-output relationship of a LAM [21] is described by

$$\mathbf{y}_r = \mathbf{W}\mathbf{p}_r + \mathbf{b} \tag{9}$$

where **W** is a weight matrix that specifies the network connectivity of the LAM, **b** is a bias vector and $\mathbf{p}_r$ is the input vector (neighborhood structure). Note that $\mathbf{y}_r$ contains a vector representation of a superresolved 2D neighborhood structure. The neighborhoods in **P** and **D** are associated in the least square sense to determine the values of the **W** and **b** parameters. These parameters can be obtained recursively via the least mean squares (LMS) algorithm update equation [21]

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \mu(\mathbf{D} - \mathbf{Y})\mathbf{P}^T \tag{10}$$

where T denotes matrix transposition and μ is the learning rate. They can also be obtained in closed form via the pseudo-inverse [21]

$$\mathbf{W} = \mathbf{D}\mathbf{P}^T\left(\mathbf{P}\mathbf{P}^T\right)^{-1}. \tag{11}$$

We have assumed in eqns. (10) and (11) that $\mathbf{W}$ is actually the augmented matrix [$\mathbf{W}$ | $\mathbf{b}$] and $\mathbf{P}$ is the augmented matrix [$\mathbf{P}^T$ | $\mathbf{v}$]^T where $\mathbf{v}$ is a column vector of ones of appropriate dimensions.

Nonlinear associative memories (NLAMs) can be used as a substitute for the LAMs of figs. (1) and (2). The parameterized nonlinear relation between the input and output can be achieved using a multilayer perceptron (MLP) [21] and is given by

$$\mathbf{y}_r = \alpha(\{\mathbf{W}_k\},\{\mathbf{b}_k\},\mathbf{p}_r) \tag{12}$$

where, in general, $\alpha(\cdot)$ is a nonlinear function of a set of weight matrices, bias vectors and the neighborhood structure and k describes a layer in the MLP feedforward configuration. The NLAM parameters are readily obtained with backpropagation learning [21].

III-*D. SUPERRESOLVING IMAGES*

The construction of a high-resolution image, as depicted in fig. (1), results from transforming the neighborhood structure of the low-resolution input image with the parameters obtained in the training phase. The mean of the neighborhood is subsequently added back to the transformation. When LAMs are used, the superresolution of a low-resolution neighborhood $\mathbf{x}_r$ can be expressed as

$$\hat{\mathbf{s}}_r = \mathbf{W}_c\mathbf{Z}\mathbf{x}_r + \mathbf{b}_c + \mathbf{A}\mathbf{x}_r \quad \text{for } (\mathbf{p}_r = \mathbf{Z}\mathbf{x}_r) \in K_c \tag{13}$$

where $\mathbf{W}_c$ and $\mathbf{b}_c$ are the weight matrix and bias vector associated with the c^{th} LAM, respectively. As discussed before, there is a direct relation between eqn. (13) and eqn. (3). Eqn. (13) constructs a high-resolution neighborhoods' structure $\hat{\mathbf{s}}_r$. The subscript r references the neighborhood being constructed. The constructed pixels that overlap are averaged and the high-resolution image is thus constructed.

Averaging several estimates improves the reliability of the final high-resolution sample. Eqn. (13) can be equivalently expressed as

$$\hat{x}_h[n_1,n_2]=\sum_{l=1}^{L}\left[x_e[n_1,n_2]**(k_{c,l}[n_1,n_2]+a[n_1,n_2])\right]\cdot b[n_1,n_2] \tag{14}$$

where $L=(2G_1-1)(2G_2-1)$, x_e is the expanded low-resolution image, the kernel was created with the values of $\mathbf{W}_c\mathbf{Z}(l,:)$ and $\mathbf{b}_c(l)$, i.e. row l of $\mathbf{W}_c\mathbf{Z}$ and $\mathbf{b}_c$, a is a constant kernel with the same extent as $k_{c,l}$ that averages a low-resolution neighborhood (its impulse response samples equal $\frac{1}{H_1H_2}$) and $b[n_1,n_2]=b_1[n_1]b_2[n_2]$ is responsible for averaging multiple estimates of superresolved samples. Specifically,

$$b_i[n_i]=\begin{cases}1 & n_i \bmod G_i=0\\ \frac{1}{2} & \text{otherwise}\end{cases} \tag{15}$$

for i = 1,2. Notice that the index l refers to a specific convolution pass that is constructing the corresponding enumerated crossed circle associated with each low-resolution neighborhood in that pass. Please refer to fig. (3a) for the case of $G_1 = G_2 = 2$.

The NLAM case only differs by the presence of nested nonlinearities. The construction, for an M layer MLP topology, is expressed as

$$\hat{\mathbf{s}}_r=\varphi\left(\mathbf{W}_{c,M}\varphi\left(\cdots\varphi\left(\mathbf{W}_{c,1}\mathbf{Z}\mathbf{x}_r+\mathbf{b}_{c,1}\right)\right)+\mathbf{b}_{c,M}\right)+\mathbf{A}\mathbf{x}_r \quad \text{for } (\mathbf{p}_r=\mathbf{Z}\mathbf{x}_r)\in K_c \tag{16}$$

where $\mathbf{W}_{c,k}$ and $\mathbf{b}_{c,k}$ are the weight matrix and bias vector at layer k of the c^{th} NLAM, respectively, and φ denotes the squashing function at each layer of the feedforward structure.

IV. RESULTS

The LAM-based results were compared against several kernel-based interpolation results including the subpixel edge localization and interpolation (SEL) technique [6], which fits an ideal step edge through those image regions where an edge was deemed to exist and otherwise uses a bilinear interpolation. The parameters for the SEL technique were the same as those reported in [6].

Table I reports on the peak signal-to-noise ratio (PSNR) resulting from kernel-based interpolation of the Lena and Peppers 128×128 images by a factor of 2 in each dimension. The PSNR is defined as $\mathrm{PSNR} \equiv -10\log_{10}(e_{rms}^2)$ where

$$e_{rms}^2 = \frac{1}{M_1 M_2} \sum_{m_1=0}^{M_1-1} \sum_{m_2=0}^{M_2-1} (x_h[m_1, m_2] - \hat{x}_h[m_1, m_2])^2 \tag{17}$$

and x_h and $\hat{x}_h$ are normalized to take values in [0,1].

Table I. PSNR for magnified images. The interpolation factor was 2 in each direction from the listed 128×128 images. The training and test cases of the kernel family approach utilized 30 features and a 3×3 ROS. In the test cases, the parameters obtained in training to reconstruct Lena where used for the Peppers and vice versa.

	Repli-cation	Bilinear	Bicubic	Cubic B spline	SEL	Train*	Test*	Train**	Test**
Lena	27.00	27.26	27.45	27.43	27.48	32.63	31.78	32.23	31.71
Peppers	27.48	27.51	27.74	27.74	27.79	34.58	32.90	34.03	32.74

* Used 30 LAMs
** Used 30 NLAMs

The plot in fig. (4) illustrates the PSNR when superresolving the Lena 128×128 image with varying number of LAMs by a factor of 2 in each dimension. The system parameters (feature vectors, weights and biases) were *trained using the Peppers 256×256 image, i.e. a different image*. This well-known image has not been included in this paper so unfamiliar readers are referred to [13,19]. The solid/dashed lines in fig. (4) denote training/test set reconstruction performance using regions of support (ROS) 3×3 and 5×5. In general, the PSNR of the training set increases as the number of LAMs increased. This is intuitively expected as an increase in the number of LAMs yields a greater specialization to particular image features, hence a more accurate image reconstruction. The feature set extracted using a 5×5 ROS yields more macroscopic image characteristics than does a 3×3 ROS. This results in greater specialization of the characteristics particular to the image of interest and generally to a more faithful image reconstruction on the training set. In the test set, however, the larger ROS tended to show a drop in

PSNR performance as the degree of specialization to image features increased. This general trend was encountered in all the tests we have run. It suggests that the similarity between features, as the system specializes more (use more LAMs), tends to occur at a more microscopic level. It can also be observed that the kernel family approach yielded higher PSNR than those methods listed in table I.

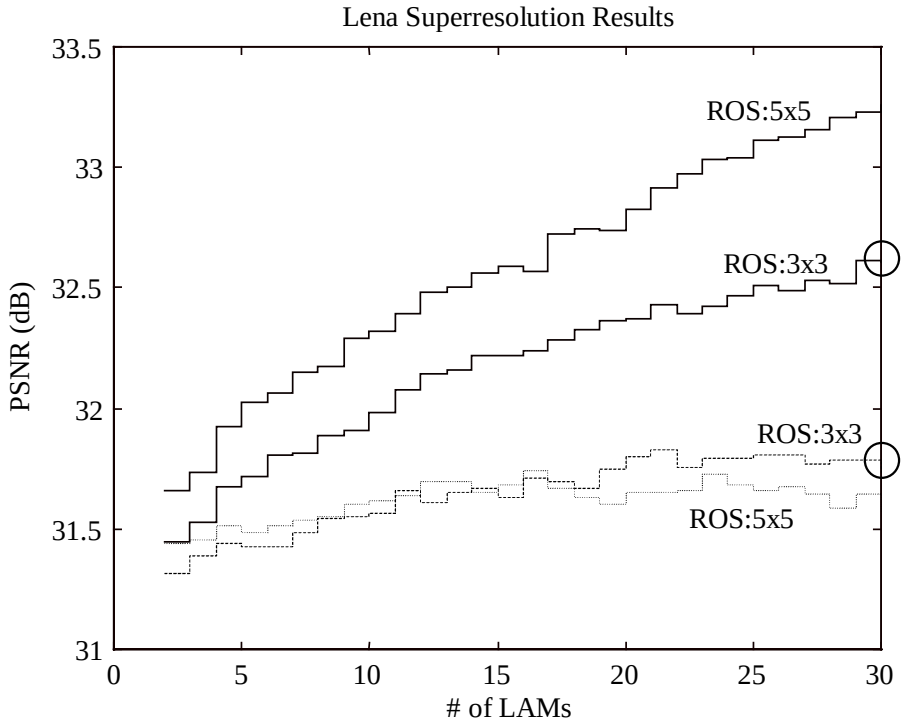


Figure 4. Training and testing superresolution results for the Lena image considering two different regions of support (ROS). The solid lines correspond to training set results and the dashed lines are test set results. The curves related to the training data result from superresolving the Lena 128×128 image with the systems (features and LAMs) trained to reconstruct the Lena 256×256 image from the Lena 128×128 image. The curves related to testing result from superresolving the Lena 128×128 image with the systems trained to reconstruct the Peppers 256×256 image from the Peppers 128×128 image. Superresolved images corresponding to the two circled points 'o' are shown in fig. (5).

A visual comparison of the results utilizing the common approaches and the kernel family approach for the Lena image can be observed in fig. (5). The training and testing images shown in each of these figures were created using 30 LAMs and a ROS of 3×3. They correspond to those points in fig. (4) marked by a circled point 'o'. The superresolved training and testing images were of similar quality. The kernel family superresolved images appear crisper than those obtained with the other approaches presented here. Edges also seemed to be preserved well with this approach.

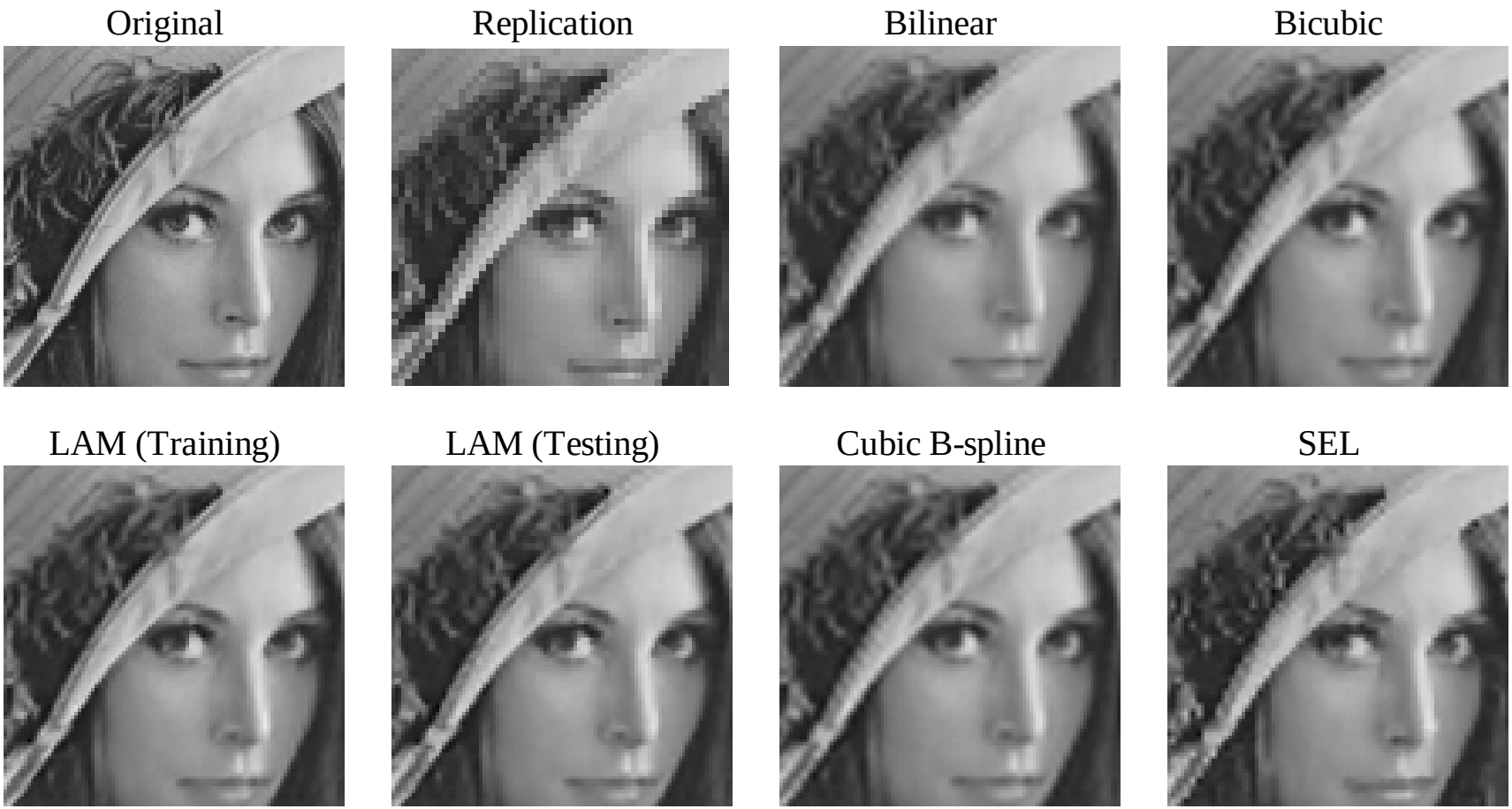


Figure 5. Visual comparison of the superresolution results for the Lena image. The 128×128 image was superresolved to 256×256. A zoomed section (using nearest neighbor replication) of the superresolved results is displayed. The 'training' parameters utilized were the 30 features and corresponding LAMs obtained in training to superresolve the Lena 256×256 image from the Lena 128×128 image with ROS of 3×3. The 'testing' parameters utilized were the 30 features and corresponding LAMs obtained in training to superresolve the Peppers 256×256 image from the Peppers 128×128 image with ROS 3×3.

Fig. (6) shows results when NLAMs are used in place of the LAMs. The NLAMs used a single hidden layer and had approximately the same number of free parameters as did the LAMs. The LAM and NLAM results are very similar (both visually and in PSNR). In this and several other tests we have run with superresolution factors of 2 and 3, the added complexity and flexibility afforded by the NLAMs seems unwarranted. This makes sense since many nonlinear mappings are reasonably well approximated locally by linear models.

LAM (Testing) NLAM (Testing)

Figure 6. Comparing the superresolution of Lena using LAMs and NLAMs. The overall PSNR performance of the LAM and NLAM based results were very similar. Here the Lena 128×128 image is superresolved by a factor of two in each image axis. The 30 features and LAMs (NLAMs respectively) used were those obtained in training to superresolve the Peppers 256×256 image from the Peppers 128×128 image with an ROS of 3×3.

As a final result to test our hypothesis that the system captures well redundancy across scales, in fig. (7) we illustrate the superresolution of the Lena 128×128 image using two successive $G_1 = G_2 = 2$ reconstruction stages using the *same codebook and LAMs*. In other words, the resulting "test" image of fig. (5) is fed through the system of fig. (1) twice with the same parameters used in the first superresolution stage for a total superresolution factor of 4 in each dimension. Notice that the LAM reconstructed image is crisper than the other expanded images. This also supports our claim regarding the similarity of image neighborhoods across scales – which we exploit for superresolution.

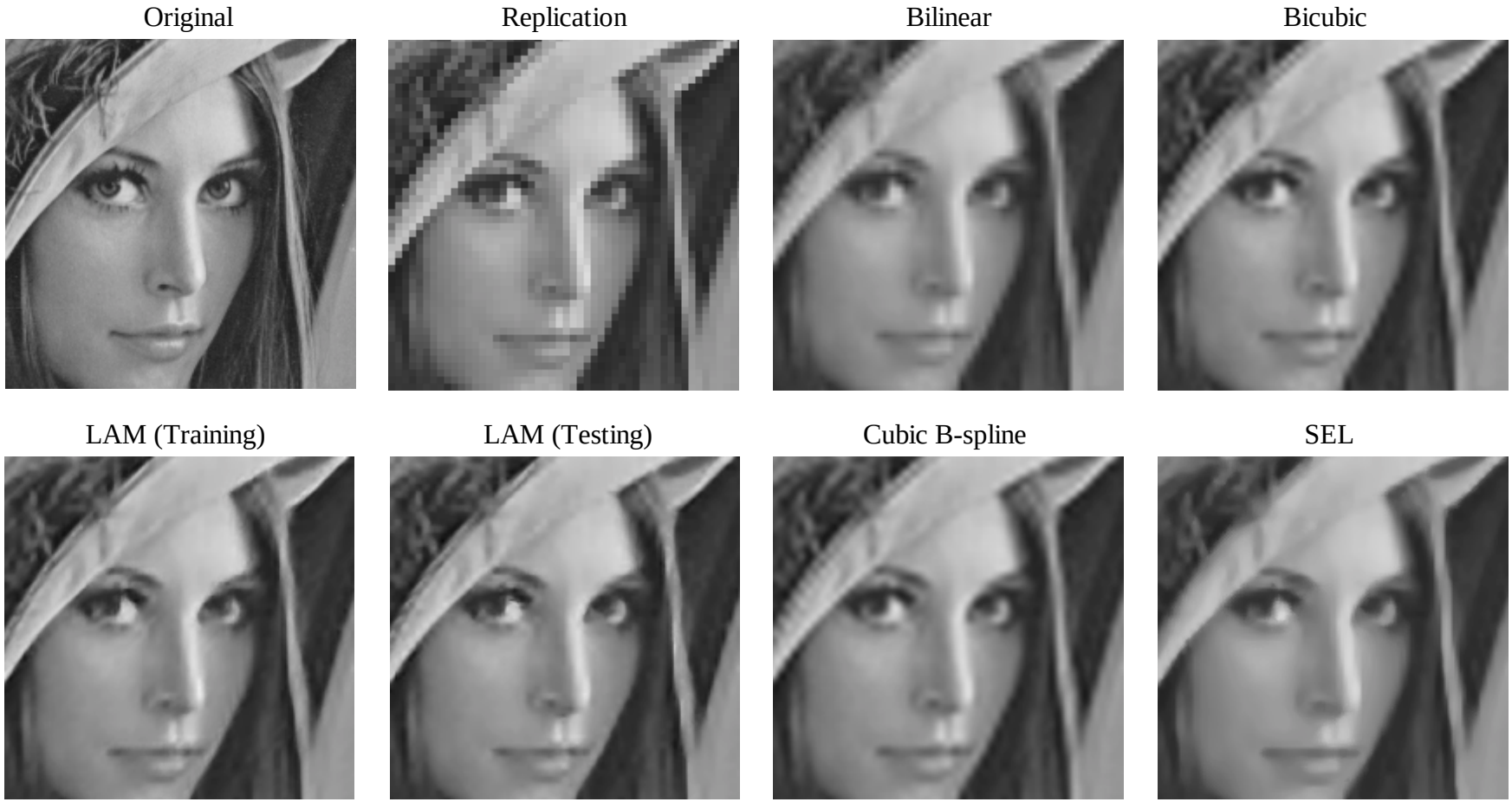


Figure 7. Example of superresolving the Lena 128×128 image by a factor of 16 (a factor of 4 along each image axis). The superresolution was accomplished using two successive reconstruction stages, each by a factor of 4. The same features and LAMs were used in each stage. The training image was superresolved using the features and LAMs obtained in superresolving the Lena 256×256 image from the Lena 128×128 image with a ROS of 3×3. The test image was superresolved using the features and LAMs obtained to superresolve the Peppers 256×256 image from the Peppers 128×128 image with a ROS of 3×3.

V. ISSUES AND NOTES

This paper reports on a novel technique for image superresolution which exploits local redundancies among and across scales. Although the preliminary results are very promising, there are many issues requiring further analysis. Noteworthy issues pertaining to the superresolution process herein are:

- The feature vectors and LAMs are established in a manner that is not driven directly by the error rate of superresolution. This is potentially suboptimal. However, because the function defining our input space partition (the clustering stage) is not differentiable this issue is not easily addressed. We have tested our approach using the hierarchical mixture of experts [22] which trains to minimize the error rate (affine experts (LAMs) and affine transformations for the gating structure were used) and our method trained faster and consistently produced higher PSNR in the reconstructed images [13].
- The topological mapping property of Kohonen's SOM was not used for the results presented here. We used the SOM because of its efficient training approach and its tendency for full codebook utilization. We have performed the clustering with the Neural Gas algorithm [23] and have not noticed performance differences [13]. The incorporation of the topological information of the SOM to improve the superresolution is the subject of current research.
- Nonlinear associative memories showed no improvement with respect to LAM performance for the parameters utilized in these experiments. Since the neighborhood sizes and the superresolution factors were small, a linear mapper seems to capture well the redundancy across scales. However, for larger superresolution factors the mapping will tend to be more and more nonlinear, so NLAMs may yield a performance advantage.
- The superresolution approach herein also allows for noninteger (rational) magnification factors. The size of the images associated (as well as what local samples are to be constructed) determines this factor for the feature vectors and LAMs established. Thus different feature and LAM sets must be established for different magnification factors.

- The low-resolution neighborhood size used is a trade-off between the amount of local support considered and how much information is to be constructed. As a rule of thumb, we suggest setting $H_i \geq 2G_i - 1$ but keeping H_i, $(i = 1,2)$, reasonably small. The number of free parameters is determined by the low-resolution support specified by H_i. Note that as G_i increases, there is more missing information to construct, hence more low-resolution sample support is needed.
- The results presented here use a single image - the Peppers - for training. However, multiple images can easily be (and have been [19]) used for training the system parameters in fig. (1). Our experiments have revealed that there is much similar local structure among images, which might not be apparent when images are casually viewed [13].
- The number of input vectors should be much larger than the dimensionality of the input space for proper LAM training. This results in the solution of an overdetermined problem rather than an underdetermined one.
- The system in fig. (1) lends itself to the incorporation of new or additional features (and LAMs) and *does not require retraining* of the existing parameters. This allows for quick amending of the bases used for reconstruction [13].

VI. CONCLUSIONS

A novel paradigm has been presented for the superresolution of optical images. The procedure was shown to be equivalent to convolution of the image with a family of kernels developed from a training image. The ill-posed superresolution problem was addressed by determining locally the optimal least squares projections across scales for image neighborhoods of similar character. The similarity among neighborhoods was characterized by their interblock correlation. The key assumption of our approach was that this similarity of neighborhoods in the low-resolution image also held across scales – an assumption that we've noticed experimentally to be very reasonable.

The use of LAMs for the local transformation is interesting in that the relation between correlated neighborhoods' structure across scales seems reasonably modeled by an affine mapping. This simplifies

the training and eases the need for establishing more complicated nonlinear transformations. Also, the need for an analysis that mathematically supports the assumptions we've observed to be reasonable is warranted and has been left for future research.

REFERENCES

[1] G. R. Benitz, "High Definition Vector Imaging for Synthetic Aperture Radar," *31st Asilomar Conf. on Signals, Sys. and Comp*., Nov. 1997

[2] A. M. Tekalp, *Digital Video Processing*, Upper Saddle River, N.J.: Prentice Hall, ch. 17, 1995.

[3] A. N. Netravali and B. G. Haskell, *Digital Pictures: Representation, Compression and Standards*, 2nd Ed., New York:Plenum Press, 1995.

[4] M. Unser, A. Aldroubi and M. Eden, "Fast B-Spline Transforms for Continuous Image Representation and Interpolation," *IEEE Trans. Pattern Anal. Mach. Int*., Vol. 13, No. 3, pp. 277-285, 1991.

[5] S. D. Bayrakeri and R. M. Mersereau, "A New Method for Directional Image Interpolation," *Proc. Int. Conf. Acoustics, Speech, Sig. Process*., Vol. 4, pp. 2383-2386, 1995.

[6] K. Jensen and D. Anastassiou, "Subpixel Edge Localization and the Interpolation of Still Images," *IEEE Trans. Image Process*., Vol. 4, No. 3, pp. 285-295, 1995

[7] A. M. Darwish and M. S. Bedair, "An Adaptive Resampling Algorithm for Image Zooming," *Proc. SPIE*, Vol. 2666, pp. 131-144, 1996.

[8] S. A. Martucci, "Image Resizing in the Discrete Cosine Transform Domain," *Proc. Int. Conf. Image Process*., Vol. 2, pp. 244-247, 1995.

[9] E. Shinbori and M. Takagi, "High Quality Image Magnification Applying the Gerchberg-Papoulis Iterative Algorithm with DCT," *Systems and Computers in Japan*, Vol. 25, No. 6, pp. 80-90, 1994.

[10] S. G. Chang, Z. Cvetkovic and M. Vetterli, "Resolution Enhancement of Images Using Wavelet Transform Extrema Extrapolation," *Proc. Int. Conf. Acoustics, Speech, Sig. Process*., Vol. 4, pp. 2379-2382, 1995.

[11] N. B. Karayiannis and A. N. Venetsanopoulos, "Image Interpolation Based on Variational Principles," *Signal Process*., Vol. 25, pp. 259-288, 1991.

[12] R. R. Schultz and R. L. Stevenson, "A Bayesian Approach to Image Expansion for Improved Definition," *IEEE Trans. Image Process*., Vol. 3, No. 3, pp. 233-242, 1994.

[13] F. M. Candocia, "A Unified Superresolution Approach for Optical and Synthetic Aperture Radar Images," Ph.D. Dissertation, Univ. of Florida, Gainesville, 1998.

[14] G. K. Wallace, "The JPEG Still Image Compression Standard," *Commun. ACM*, Vol. 34, No. 4, pp. 30-44, 1991.

[15] R. D. Dony and S. Haykin, "Neural Network Approaches to Image Compression," *Proc. IEEE*, Vol. 83, No. 2, pp. 288-303, 1995.

[16] R. D. Dony and S. Haykin, "Optimally Integrated Adaptive Learning," *IEEE Trans. Image Proc*., Vol. 4, No. 10, pp. 1358-1370, 1995.

[17] N. Kambhatla and T. Leen, "Dimension Reduction by Local Principal Component Analysis", *Neural Computation*, Vol. 9, pp. 1493-1516, 1997.

[18] R. W. Schafer and L. R. Rabiner, "A Digital Signal Processing Approach to Signal Interpolation," *Proc. IEEE*, Vol. 61, No. 6, pp. 692-702, 1973.

[19] F. M. Candocia and J. C. Principe, "A Neural Implementation of Interpolation with a Family of Kernels," *Proc. Int. Conf. Neural Net*., Vol. 3, pp. 1506-1510, 1997.

[20] T. Kohonen, "The Self-Organizing Map," *Proc. IEEE*, Vol. 78, pp. 1464-1480, 1990.

[21] S. Haykin, *Neural Networks: A Comprehensive Foundation*, New York:MacMillan, 1994.

[22] M. I. Jordan and R. A. Jacobs, "Hierarchical Mixtures of Experts and the EM Algorithm," *Neural Computation*, Vol. 6, pp. 181-214, 1994.

[23] T. M. Martinez, S. G. Berkovich and K. J. Schulten, " 'Neural Gas' Network for Vector Quantization and its Applications to Time-Series Prediction," *IEEE Trans. Neural Networks*, Vol. 4, No. 4, pp. 558-569, 1993.