THE VECTOR HYBRID WIENER FILTER: APPLICATION TO SUPER-RESOLUTION

Tomer Michaeli and Yonina C. Eldar

Department of Electrical Engineering
Technion–Israel Institute of Technology, Haifa, Israel
{tomermic@tx, yonina@ee}.technion.ac.il

ABSTRACT

We address the problem of recovering a continuous-time (space) signal from several blurred and noisy sampled versions of it, a scenario commonly encountered in super-resolution (SR) and array-processing. We show that discretization, a key step in many SR algorithms, inevitably leads to inaccurate modeling. Instead, we treat the problem entirely in the continuous domain by modeling the signal as a continuous-time random process and deriving its linear minimum mean-squared error estimate given the samples. We also provide an efficient implementation scheme, valid for 1D applications. Simulation results on real-world data demonstrate the advantage of our approach with respect to SR techniques that rely on discretization.

Index Terms— Super-resolution, nonuniform interpolation.

1. INTRODUCTION

In many applications, multiple noisy discrete-time (space) observations of a continuous-time (space) signal are available. These scenarios can often be regarded as multichannel sampling problems, where each channel outputs uniformly-spaced samples of a filtered version of the signal, as depicted in Fig. 1. The goal in these situations is to recover the original signal $x(t)$ from the K sequences $\{c_k[n]\}_{k=1}^K$. One extensively studied application of this type of problem is super-resolution (SR) from image sequences [1]. Here, several low-resolution noisy images of a scene are captured by a camera, each with a different translation. An SR algorithm is targeted at combining these images into one high-resolution image of the same scene. In this case, the sampling filters in Fig. 1 are given by $s_k(t) = s(t - t_k)$, where t is a 2D coordinate vector, $s(t)$ is the point-spread-function (PSF) of the imaging device and t_k is the translation of the kth frame. One-dimensional versions of the scheme in Fig. 1 arise in beamforming [2] and in temporal SR applications [3]. In the latter, several video sequences of the same scene are processed to produce one video stream at a higher frame-rate. In the sequel, we collectively refer to all problems of this type as SR.

In much of the recent literature, SR is modeled via the finite-dimensional discrete-time (space) relations

$$c_k = S_k x_{\text{HR}} + u_k, \qquad k = 1, \dots, K. \tag{1}$$

Here c_k is a vector containing the available samples from the kth channel, x_{HR} is a vector comprised of the samples of $x(t)$ on the desired dense grid, u_k is a noise vector, and S_k is a matrix which accounts for the filtering and sampling operations in Fig. 1. As we
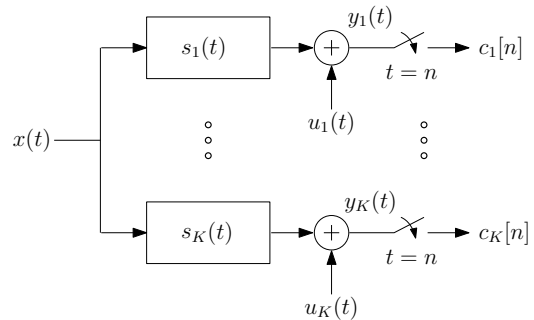
Fig. 1: Multichannel sampling scheme.

show in this paper, there are many situations in which such a representation is not possible. Thus, treatment in the continuous domain is required, namely addressing the problem from an interpolation viewpoint.

Prior SR work, falling into the interpolation category, include [4, 5, 6, 7, 8]. These methods suffer from two major drawbacks. First, the original signal $x(t)$ is assumed to be bandlimited. It is well known that this type of prior knowledge is not in agreement with the typical behavior of natural images. Furthermore, no statistical assumptions on $x(t)$ are incorporated, which have been shown to greatly improve SR performance when using the discrete model (1) [1]. An exception in this regard is the work in [8], where prior knowledge on $x(t)$ is implicitly defined by a user-chosen "back-projection kernel". However, the effect of this kernel on the reconstruction is not clear. A second drawback is that all these methods are iterative, and thus are not well suited for real-time implementation.

In this paper we treat the SR interpolation problem without resorting to the standard bandlimited assumption. Motivated by the good single-channel image interpolation results recently reported in [9], here we model $x(t)$ as a wide-sense-stationary (WSS) random signal with known power-spectral-density (PSD). We then derive the linear minimum mean-squared error (MSE) estimator of $x(t)$ given the sequences $\{c_k[n]\}_{k=1}^K$. The resulting algorithm is highly efficient: it merely consists of digital filtering of the sequences $c_k[n]$, followed by simple interpolation. We present closed form expressions for both stages and also an efficient digital filtering method, valid for 1D scenarios.

2. THE NEED FOR CONTINUOUS-TIME TREATMENT

We begin by studying the situations in which the multichannel sampling problem can be accurately treated using the discrete form (1). For concreteness, we assume in this section that $s_k(t) = s(t - t_k)$

for a set of translations $\{t_k\}_{k=1}^K$. We wish to relate the sequences $\{c_k[n]\}_{k=1}^K$ to a densely-sampled version of $x(t)$, modeled as

$$x_{\text{HR}}[n] = (s(\Delta t) * x(t))|_{t=n/\Delta}, \qquad (2)$$

where $\Delta > 1$ is the magnification factor.

Both $x_{\text{HR}}[n]$ and $c_k[n]$ can be expressed as inner products with a set of functions:

$$c_k[n] = \langle x(t), s(n - t_k - t) \rangle, \; x_{\text{HR}}[n] = \langle x(t), s(n - \Delta t) \rangle. \quad (3)$$

Expressing $c_k[n]$ as a linear combination of $\{x_{\text{HR}}[m]\}_{m \in \mathbb{Z}}$, we have

$$
\begin{aligned}
c_k[n] &= \sum_{m \in \mathbb{Z}} \boldsymbol{S}_k[n,m] x_{\text{HR}}[m] \\
&= \sum_{m \in \mathbb{Z}} \boldsymbol{S}_k[n,m] \langle x(t), s(m - \Delta t) \rangle \\
&= \left\langle x(t), \sum_{m \in \mathbb{Z}} \boldsymbol{S}_k[n,m], s(m - \Delta t) \right\rangle. \qquad (4)
\end{aligned}
$$

Therefore, from (3) we conclude that such a representation is possible only if for every $n \in \mathbb{Z}$, there exists a sequence $\{\boldsymbol{S}_k[n,m]\}_{m \in \mathbb{Z}}$ such that

$$s(t + n - t_k) = \sum_{m \in \mathbb{Z}} \boldsymbol{S}_k[n,m] s(\Delta t + m). \qquad (5)$$

If, for example, $s(t)$ is a rectangular window of width 1, as frequently happens in CCD cameras, then (5) cannot be satisfied unless the magnification factor Δ is an integer and the translation t_k is an integer multiple of $1/\Delta$. In fact, these two last assumptions are commonly made in SR algorithms. However, they are not sufficient to guarantee the satisfaction of (5) for other types of filters. The next theorem provides a frequency-domain characterization of the filters that satisfy (5) in this setting.

Theorem 1 *Assume that Δ is an integer and that t_k is an integer multiple of $1/\Delta$. Then for every n, $c_k[n]$ of Fig. 1 can be expressed as a linear combination of $\{x_{HR}[m]\}_{m \in \mathbb{Z}}$ of (2) only if there exists a $2\pi\Delta$-periodic function $A(e^{j\omega/\Delta})$ such that*

$$S(\omega) = S\left(\frac{\omega}{\Delta}\right) A\left(e^{j\frac{\omega}{\Delta}}\right) \qquad (6)$$

for every $\omega \in \mathbb{R}$. Here $S(\omega)$ denotes the Fourier transform of $s(t)$.

As an example, it is easily verified that the widely used Gaussian PSF model does not satisfy (6) and therefore cannot be treated via the discrete model (1). We conclude that to be loyal to the physical setting, SR must be addressed in the continuous domain.

3. THE VECTOR HYBRID WIENER FILTER

Although the sequences $\{c_k[n]\}_{k=1}^K$ cannot be expressed as linear transformations of $x_{\text{HR}}[n]$, we can still derive an estimate of the latter based on the former. To do so, we will first estimate the continuous-time signal $s(\Delta t) * x(t)$ from the channel outputs, and then sample it on a grid with $1/\Delta$ spacings. Note that the linear minimum MSE (LMMSE) estimate of $s(\Delta t) * x(t)$ is simply $s(\Delta t) * \hat{x}(t)$, where $\hat{x}(t)$ is the LMMSE estimate of $x(t)$. Therefore, we now address the recovery of the continuous-time signal $x(t)$ from the discrete-time channel outputs.

Our goal is to *linearly* estimate $x(t)$ given the equidistant point-wise samples of the K signals $\{y_k(t)\}_{k=1}^K$ such that the MSE

$E[(x(t) - \hat{x}(t)^2]$ is minimized for every t. To make the derivation general, all we assume in this section is that $x(t)$ and $\{y_k(t)\}_{k=1}^K$ are jointly WSS. We denote their cross-correlation functions by $R_{xy}^k(\tau) = E[x(t)y_k(t - \tau)]$ and $R_{yy}^{k,\ell}(\tau) = E[y_\ell(t)y_k(t - \tau)]$. The cross-spectra are given by the Fourier transforms $\Gamma_{xy}^k(\omega) = \mathcal{F}\{R_{xy}^k\}$ and $\Gamma_{yy}^{k,\ell}(\omega) = \mathcal{F}\{R_{yy}^{k,\ell}\}$.

We wish to construct an estimate of the form

$$\hat{x}(t) = \sum_{k=1}^K \sum_{n \in \mathbb{Z}} c_k[n] w_k(t - n). \qquad (7)$$

The case where only one measurement channel is available (*i.e.*, $K = 1$) was treated in [10]. The resulting reconstruction formula is referred to as the scalar hybrid Wiener filter since its input is the (scalar) discrete-time signal $y_1(n)$, $n \in \mathbb{Z}$, whereas its output is a continuous-time signal $\hat{x}(t)$, $t \in \mathbb{R}$. Consequently, we refer to our multichannel setup (for $K > 1$) as the vector hybrid Wiener filter.

Theorem 2 (Vector hybrid Wiener filter) *The reconstruction kernels $\{w_k(t)\}_{k=1}^K$ in (7) that minimize the MSE are given in the frequency domain by*

$$\boldsymbol{W}(\omega) = \left(\sum_{n \in \mathbb{Z}} \boldsymbol{\Gamma}_{yy}(\omega - 2\pi n) \right)^{-1} \boldsymbol{\Gamma}_{xy}(\omega), \qquad (8)$$

where $\boldsymbol{W}(\omega) = (W_1(\omega), \dots, W_K(\omega))^T$, $\boldsymbol{\Gamma}_{xy}(\omega) = (\Gamma_{xy}^1(\omega), \dots, \Gamma_{xy}^K(\omega))^T$, and $\boldsymbol{\Gamma}_{yy}(\omega)$ is a $K \times K$ matrix whose (k, ℓ) entry is $\Gamma_{yy}^{k,\ell}(\omega)$.

The proof of the theorem relies on the orthogonality principle, following similar lines to [10], and is omitted due to lack of space.

4. EFFICIENT IMPLEMENTATION SCHEME

We now specialize the hybrid Wiener filter to the SR setting and provide an efficient implementation scheme for the case where both the PSF and the signal's autocorrelation are of finite support.

Using the relation $\mathcal{F}\{\sum c[n]h(t - n)\} = C(e^{j\omega})H(\omega)$, it can be shown that the reconstruction formula (7) can be implemented in two stages. First, the vector process $\boldsymbol{c}[n] = (c_1[n], \dots, c_K[n])^T$ is digitally filtered by the $K \times K$ MIMO digital filter

$$\boldsymbol{H}\left(e^{j\omega}\right) = \left(\sum_{n \in \mathbb{Z}} \boldsymbol{\Gamma}_{yy}(\omega - 2\pi n) \right)^{-1} \qquad (9)$$

to obtain a "corrected" process $\boldsymbol{d}[n] = (d_1[n], \dots, d_K[n])^T$. Then, $\hat{x}(t)$ is formed using

$$\hat{x}(t) = \sum_{k=1}^K \sum_{n \in \mathbb{Z}} d_k[n] v_k(t - n), \qquad (10)$$

with the kernels $v_k(t) = R_{xy}^k(t)$. This scheme is depicted in Fig. 2.

In order to specialize this scheme to the SR setting, we now make the following assumptions:

1. $s_k(t) = s(t - t_k)$ for a set of translations $\{t_k\}_{k=1}^K$ in the range[1] $[-0.5, 0.5]$,

2. $\{u_k(t)\}_{k=1}^K$ are independent of $x(t)$ and are characterized by $R_{uu}^{k,\ell}(\tau) = E[u_k(t)u_\ell(t - \tau)] = \sigma_u^2 \delta_{k,\ell} \text{sinc}(\tau)$,

---

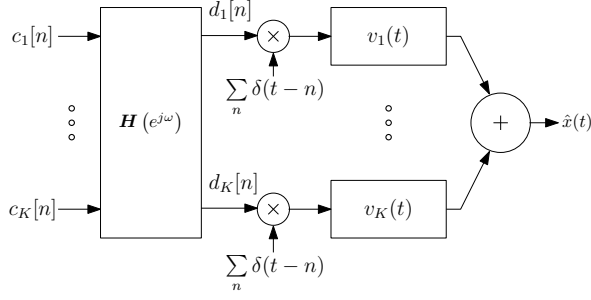[1] A shift $|t_k| > 0.5$ can be handled by pre-introducing an integer shift of $-\lfloor t_k + 0.5 \rfloor$ samples to $c_k[n]$.

Fig. 2: Multichannel reconstruction scheme.

3. supp$\{s(t)\} \subseteq [-L_s, L_s]$, and
4. supp$\{R_{xx}(t)\} \subseteq [-L_x, L_x]$.

Assumption 2 is equivalent to replacing the analog noise signals $\{u_k(t)\}_{k=1}^K$ by digital white noise processes $\{u_k[n]\}_{k=1}^K$, which are added after the sampling occurs. Assumptions 3 and 4 are required in order to make the recovery algorithm practical, as we detail next.

We first examine the interpolation stage in Fig. 2. It can be shown that under assumptions 1 and 2,

$$v_k(t) = R_{xy}^k(t) = (\bar{s} * R_{xx})(t + t_k), \qquad (11)$$

where we denoted $\bar{s}(t) = s(-t)$. Furthermore, assumptions 3 and 4 imply that the supports of the kernels $\{v_k(t)\}_{k=1}^K$ are finite, making the interpolation practical. Specifically, for any t_0, the computation of $\hat{x}(t_0)$ involves only $\lfloor 2(L_s + L_x) \rfloor$ multiplications in each of the K branches. Note that to estimate $s(\Delta t) * x(t)$, we need to use the interpolation kernels $\{s(\Delta t) * v_k(t)\}_{k=1}^K$, which are also compactly supported. If Δ is large, this modification can be neglected.

Next, we show that the MIMO digital filtering stage can also be implemented efficiently. Indeed, the filter $\boldsymbol{H}(e^{j\omega})$ of (9) is the convolutional inverse of the matrix sequence $\boldsymbol{Q}[n]$, whose (k, ℓ) entry is $R_{yy}^{k,\ell}(n)$. In our setting, $R_{yy}^{k,\ell}(t)$ is given by

$$R_{yy}^{k,\ell}(t) = (\bar{s} * s * R_{xx})(t + t_k - t_\ell) + R_{uu}^{k,\ell}(t), \qquad (12)$$

and therefore $\boldsymbol{Q}_{k,\ell}[n] = (\bar{s} * s * R_{xx})(n + t_k - t_\ell) + \sigma_u^2 \delta_{k,\ell}\delta[n]$. Assumptions 3 and 4 imply that $R_{yy}^{k,\ell}(t)$ is compactly supported and thus $\boldsymbol{Q}[n]$ is a finite sequence whose Z-transform can be written as

$$\boldsymbol{Q}(z) = \boldsymbol{A}_p^T z^{-p} + \ldots + \boldsymbol{A}_1^T z^{-1} + \boldsymbol{A}_0 + \boldsymbol{A}_1 z + \ldots + \boldsymbol{A}_p z^p, \ (13)$$

where $p = \lceil 2L_s + L_x \rceil$. The key observation is that if $\boldsymbol{Q}(z)$ is positive definite on the unit circle $|z| = 1$, then it can be factored as

$$\boldsymbol{Q}(z) = \boldsymbol{B}(z)\boldsymbol{B}^T(z^{-1}), \qquad (14)$$

where

$$\boldsymbol{B}(z) = \boldsymbol{B}_0 + \boldsymbol{B}_1 z + \ldots + \boldsymbol{B}_p z^p \qquad (15)$$

is a matrix whose determinant does not vanish inside the unit circle $|z| \leq 1$. Such a factorization can be obtained *e.g.,* by any one of the methods surveyed in [11]. Armed with (14), we can now carry out the filtering by $\boldsymbol{H}(e^{j\omega})$ of (9) in two stages. First, we form an auxiliary sequence $\boldsymbol{c}'[n]$ using the stable recursive formula

$$\boldsymbol{c}'[n] = \boldsymbol{B}_0^{-1}\left(\boldsymbol{c}[n] - \boldsymbol{B}_1\boldsymbol{c}'[n+1] - \ldots - \boldsymbol{B}_p\boldsymbol{c}'[n+p]\right) \quad (16)$$

running from right to left. Then, we form $\boldsymbol{d}[n]$ using the stable filter

$$\boldsymbol{d}[n] = \boldsymbol{B}_0^{-T}\left(\boldsymbol{c}'[n] - \boldsymbol{B}_1^T\boldsymbol{d}[n-1] - \ldots - \boldsymbol{B}_p^T\boldsymbol{d}[n-p]\right) \quad (17)$$

running from left to right.

The technique outlined above follows similar lines to the direct B-spline transform introduced in [12], which is used for spline interpolation from uniformly spaced samples. There are, however, two major differences a practitioner must be aware of, which are caused by the fact that the SR setting is more complicated. First, in contrast to the scalar interpolation scenario, here the order in which (16) and (17) are performed is important. Second, our scheme cannot be extended to multiple dimensions by operating along each dimension separately, as done in the scalar case. Specifically, even if the PSF $s(\boldsymbol{t})$ and the autocorrelation $R_{xx}(\boldsymbol{t})$ are separable functions of the coordinates $\boldsymbol{t}$, the digital filtering by $\boldsymbol{H}(e^{j\omega})$ of (9) is generally not equivalent to applying (16) and (17) sequentially on each dimension. Algorithm 1 summarizes the SR interpolation scheme devised above for 1D signals.

Algorithm 1 Fast 1D hybrid Wiener super-resolution.

Input: Samples $\{c_k[n]\}_{k=1}^K$, noise variance σ_u^2, shifts $\{t_k\}_{k=1}^K$ in the range $[-0.5, 0.5]$, PSF $s(t)$ supported on $[-L_s, L_s]$, signal's autocorrelation $R_{xx}(t)$ supported on $[-L_x, L_x]$.
Output: Recovery $\hat{x}(t)$.
1: **PSD computation:** Set $(\boldsymbol{A}_0)_{k,\ell} = (\bar{s}*s*R_{xx})(t_k - t_\ell) + \sigma_u^2\delta_{k,\ell}$ and $(\boldsymbol{A}_n)_{k,\ell} = (\bar{s}*s*R_{xx})(n + t_k - t_\ell)$, $n = 1, \ldots, p$, where $p = \lceil 2L_s + L_x \rceil$.
2: **Spectral factorization:** Given $\{\boldsymbol{A}_k\}_{k=0}^p$, compute matrices $\{\boldsymbol{B}_k\}_{k=0}^p$ satisfying (14) using any matrix spectral factorization algorithm *e.g.,* one of the methods in [11].
3: **Noncausal filtering:** Apply (16) on the vector process $\boldsymbol{c}[n] = (c_1[n], \ldots, c_K[n])^T$ to obtain $\boldsymbol{c}'[n]$.
4: **Causal filtering:** Apply (17) on $\boldsymbol{c}'[n]$ to obtain the vector process $\boldsymbol{d}[n] = (d_1[n], \ldots, d_K[n])^T$.
5: **Interpolation:** Compute $\hat{x}(t)$ using (10) with $\{v_k(t)\}$ of (11).

4.1. Spline super-resolution

In practical scenarios, the PSD $\Gamma_{xx}(\omega)$ should be chosen to roughly match the typical frequency content of the signals encountered in a specific application. On the other hand, it is desired from a computational perspective, that the autocorrelation function $R_{xx}(t)$ have a small support. An attractive choice, which has been shown to be well suited for natural images, is to let $R_{xx}(t)$ be a B-spline of degree D_x, say 2 or 3. A B-spline of degree N, denoted $\beta_N(t)$ is the function obtained by the $(N + 1)$-fold convolution of the unit square $\beta_0(t) = \mathbf{1}_{[-0.5,0.5]}(t)$. If we also model the PSF $s(t)$ as a B-spline of some degree, say D_s, then from (11) we see that the interpolation kernels $\{v_k(t)\}_{k=1}^K$ are shifted versions of B-splines of degree $D_x + D_s + 1$. These functions are compactly supported and require only $D_x + D_s + 2$ multiplications per branch in Fig. 2 to compute $\hat{x}(t)$ for any given t. We term the resulting scheme *spline super-resolution* (SSP).

5. SIMULATIONS

5.1. 2D Example

Figure 3 shows an example of SR from an image sequence. In this experiment, we used the first $K = 20$ frames from the sequence Disk taken from [13] to produce an image whose resolution is $\Delta = 4$ times higher. The PSF was modeled as a Gaussian window with variance 1. The 2D PSD was chosen as $\Gamma_{xx}(\boldsymbol{\omega}) =$

(a) One frame from low-resolution sequence.

(b) Super-resolved image using [14].

(c) Hybrid Wiener super-resolved image.

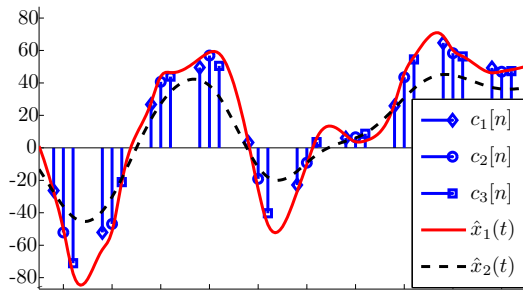**Fig. 3**: 2D super-resolution from an image sequence.

Fig. 4: 1D interpolation using the SSP algorithm.

$(\|\boldsymbol{\omega}\|^2 + (0.05\pi)^2)^{-1.5}$, to account for the typical polynomial falloff of the frequency content in natural images. The noise variance was tuned to yield input SNR of 7dB. In this 2D experiment we applied the digital filtering stage using the discrete-time Fourier transform (DTFT) by computing $\boldsymbol{H}(\boldsymbol{\omega})$ of (9) for a discrete set of frequencies. To appreciate the importance of correct modeling, we compared our result to the ℓ_1-regularized robust SR algorithm of [14], which is among the prominent SR techniques that rely on discrete formulation. We used the same estimate for the translations in both algorithms. As can be seen in Fig. 3, our approach better reconstructs the text (note in particular the letters 'R' 'E' and 'F').

5.2. 1D Example

Figure 4 depicts two 1D interpolation examples produced by the SSP algorithm. Here, the PSF $s(t)$ is the rectangular window $\beta_0(t)$, $R_{xx}(t) = \beta_2(t)$, and the shifts are $(t_1, t_2, t_3) = (-0.1, 0, 0.1)$. The solid curve is the recovery $\hat{x}(t)$ obtained when setting $\sigma_u^2 = 0$. This reconstruction is consistent, namely its samples coincide with the sequences $\{c_k[n]\}_{k=1}^K$. The dashed curve is the estimate obtained when setting $\sigma_u^2 = 1$. It can be seen that in this case, $\hat{x}(t)$ tends to be smoother at the cost of deviating from the measured samples.

6. REFERENCES

[1] S. Farsiu, D. Robinson, M. Elad, and P. Milanfar, "Advances and challenges in super-resolution," *Int. J. Imaging Syst. Technol.*, vol. 14, no. 2, 2004.

[2] R. G. Pridham and R. A. Mucci, "Digital interpolation beam-forming for low-pass and bandpass signals," *Proceedings of the IEEE*, vol. 67, no. 6, pp. 904–919, 1979.

[3] E. Shechtman, Y. Caspi, and M. Irani, "Increasing space-time resolution in video," *Lecture Notes in Computer Science*, pp. 753–768, 2002.

[4] K. Aizawa, T. Komatsu, and T. Saito, "A scheme for acquiring very high resolution images using multiplecameras," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing.*, vol. 3, 1992.

[5] K. Sauer and J. Allebach, "Iterative reconstruction of bandlimited images from nonuniformly spaced samples," *IEEE Trans. Circuits Syst.*, vol. 34, no. 12, pp. 1497–1506, 1987.

[6] H. Shekarforoush and R. Chellappa, "Data-driven multichannel superresolution with application to video sequences," *J. Opt. Soc. Am. A*, vol. 16, no. 3, pp. 481–492, 1999.

[7] A. M. Tekalp, M. K. Ozkan, and M. I. Sezan, "High-resolution image reconstruction from lower-resolution imagesequences and space-varying image restoration," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing.*, vol. 3, 1992.

[8] M. Irani and S. Peleg, "Improving resolution by image registration," *CVGIP: Graphical Models and Image Processing*, vol. 53, no. 3, pp. 231–239, 1991.

[9] Y. C. Eldar and T. Michaeli, "Beyond bandlimited sampling: nonlinearities, smoothness and sparsity," to appear in *IEEE Signal Process Mag.*

[10] M. B. Matthews, "On the linear minimum-mean-squared-error estimation of an undersampled wide-sense stationary random process," *IEEE Trans. Signal Process.*, vol. 48, no. 1, pp. 272–275, 2000.

[11] V. Kucera, "Factorization of rational spectral matrices: a survey of methods," in *Proc. IEE Int. Conf. Contr., Edinburgh, U.K.*, 1991, pp. 1074–1078.

[12] M. Unser, A. Aldroubi, and M. Eden, "Fast B-spline transforms for continuous image representation andinterpolation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 13, no. 3, pp. 277–285, 1991.

[13] "http://www.soe.ucsc.edu/~milanfar/software/sr-datasets. html."

[14] S. Farsiu, M. D. Robinson, M. Elad, and P. Milanfar, "Fast and robust multiframe super resolution," *IEEE Trans. Image Process.*, vol. 13, no. 10, pp. 1327–1344, 2004.