Superresolution Restoration of an Image Sequence: Adaptive Filtering Approach

Michael Elad, *Member, IEEE,* and Arie Feuer, *Senior Member, IEEE*

***Abstract*—This paper presents a new method based on adaptive filtering theory for superresolution restoration of continuous image sequences. The proposed methodology suggests least squares (LS) estimators which adapt in time, based on adaptive filters, least mean squares (LMS) or recursive least squares (RLS). The adaptation enables the treatment of linear space and time-variant blurring and arbitrary motion, both of them assumed known. The proposed new approach is shown to be of relatively low computational requirements. Simulations demonstrating the superresolution restoration algorithms are presented.**

***Index Terms*— Adaptive filters, least mean squares, recursive least squares, regularization, restoration, superresolution.**

I. Introduction

SIGNAL restoration—linear deblurring and noise suppression—is widely treated in the literature for a variety of applications [1]–[3]. Single image restoration has become a classic chapter in image processing theory [1], with a direct generalization to the restoration of continuous image sequences [2], [3]. The term *continuous* corresponds to the basic assumption that the image sequence contains one filmed scene. A standard video camera can be viewed as a source for such signals. Suppressing an additive noise in a continuous image sequence is an important preprocessing stage in many applications such as image sequence coding and computer vision algorithms. Deblurring such a signal is an important tool for the enhancement of visual data before presentation to the human viewer.

Despite its importance, this restoration problem has not been as extensively treated as the single image counterpart problem. We believe that this fact could partly be explained by the amount of computations and memory required when applying even the simplest restoration algorithm to such signals. The existing restoration methods for image sequences [2]–[6] are those with a basic simplicity which reduces them to having reasonable computational and memory requirements, traded-off by performance. The simplicity is obtained by assuming stationarity, causality in the image plane, and locality of the filter applied in various combinations. Among these algorithms are the three-dimensional (3-D) local Kalman filter [4], 3-D median filter [5] and motion-compensated simple averaging in the time axis [2].

The richness of the concerned signal—continuous image sequence—affords a new and unique possibility: Restoring the signal with an improved resolution. This idea is based on the fact that the filmed data is measured for each image in a different position (either because of camera motion or the motion of objects), and thus, several images can be combined to create an enhanced resolution output image. This idea has recently attracted attention and several algorithms have been proposed to estimate the so-called superresolution image from a given continuous image sequence [7]–[13]. Note that the superresolution problem is highly connected to the restoration issue presented earlier [13].

There are several known methods applicable to the problem of generating a superresolution image [7]–[13]. All these methods treat the problem of reconstructing one superresolution image from several warped, blurred, downsampled and noisy versions of it. We refer to this problem as the *classic* superresolution problem. Generally speaking, the above methods can be divided into two major groups. The first group—[7]–[9]—consists of the methods that must rely on several limiting assumptions such as linear space invariant (LSI) blur, global translational motion, white noise, and more. The remaining methods—[10]–[13]—can principally treat the most general case of LSV blur, nonwhite and nonhomogeneous additive noise, arbitrary geometric warp, and different measured resolutions. In [13], a unified approach toward the classic superresolution problem is proposed, which generalizes the methods proposed in [10] and [11], and connects them to the well-established theories of restoration and reconstruction. This paper is based on the above unified results.

Generalizing the approach suggested in [13], we can suggest a time and space filter operating on the given image sequence and producing, as an output, a restored *sequence of images* of higher resolution. The proposed approach enables both restoration from a linear blurring and noise suppression, together with superresolution image reconstruction. This methodology is based on the application of adaptive filtering theory in the time axis. This idea can be applied using either the least mean squares (LMS) or the recursive least squares (RLS) algorithms [14], [15]. The new approach is shown to be capable of treating any chosen output resolution, linear time- and space-variant blur, and motion flow. This concept gives progressive estimation of the output image sequence at a higher resolution. Since the proposed methodology is a direct generalization of

Manuscript received January 1, 1996; revised April 8, 1998. This work was supported by the Israel Science Foundation founded by the Israel Academy of Sciences and Humanities, and by the Technion V. P. R. Fund, N. Haar and R. Zinn Research Fund. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Jun Zhang.

M. Elad is with HP Laboratories–Israel, Technion City, Haifa 32000, Israel (e-mail: elad@hpli.hpl.hp.com).

A. Feuer is with the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel (e-mail: feuer@ee.technion.ac.il).

Publisher Item Identifier S 1057-7149(99)01567-5.

the classic superresolution problem, the use of regularization is as important here as it is for the classic superresolution problem.

Regarding the computational complexity of the proposed restoration algorithms, we show in this paper that the LMS and the RLS versions used in our algorithms require an order of L^2 operations per one output image, where L^2 is the number of required output pixels in each image. As such, this result implies that the proposed methodology results in computationally very efficient algorithms.

The paper is organized as follows. Section II briefly discusses the classic superresolution problem and the proposed reconstruction algorithms as given in [13]. Section III presents a new model for the task of restoration of a continuous image sequence with an improved resolution, based on the restoration method shown in [13]. This model motivates the construction of the new methodology and its application using LMS, and pseudo-RLS algorithms. Section IV gives a review of the computational complexity of the proposed algorithms. Simulations and analysis of the LMS and pseudo-RLS algorithms are given in Section V. Section VI summarizes the results of this paper.

II. The Classic Superresolution Problem

In this section, we give a short review of the classic superresolution problem and its solution. Further details can be found in [13]. Given are N low-resolution images of the same scene, $\underline{Y}_i$ $1 \leq i \leq N$, each containing M_i^2 pixels. The images are represented as vectors, ordered column-wise lexicographically for the convenience of matrix representation. The desired superresolution image, $\underline{X}$, contains L^2 pixels, where typically $L \geq M$. Equality $L = M$ corresponds to simple simultaneous restoration from several measurements, producing the same output resolution. Let us assume we know the blur function and the geometric warp function between each measured image and arbitrary reference image. Then, the measured images are assumed to be driven from a source image through the following equations:

$$\underline{Y}_i = D_i H_i F_i \underline{X} + \underline{E}_i \quad \text{for } 1 \leq i \leq N \tag{1}$$

where the involved matrices and vectors are as follows.

1) D_i represents the decimation operator (size $[M_i^2 \times L^2]$).
2) H_i represents the space-variant blur matrix (size $[L^2 \times L^2]$).
3) F_i represents the geometric warp matrix, (size $[L^2 \times L^2]$).
4) $\underline{E}_i$ represents an additive random noise, with autocorrelation matrix W_i^{-1} (size $[M_i^2 \times M_i^2]$). We further assume that the noise between different measurements is uncorrelated $(E[E_i E_j^T] = 0, i \neq j)$.

The decimation, the blur, the warp and the autocorrelation matrices are assumed known (see [13] for detailed description of this assumption). The superresolution estimated image $\underline{\hat{X}}$ can be obtained by minimizing the weighted squared error:

$$\epsilon^2 = ||S\underline{\hat{X}}||_V^2 + \sum_{i=1}^{N} ||\underline{Y}_i - D_i H_i F_i \underline{\hat{X}}||_{W_i}^2 = \underline{\hat{X}}^T S^T V S \underline{\hat{X}} + \sum_{i=1}^{N} (\underline{Y}_i - D_i H_i F_i \underline{\hat{X}})^T W_i (\underline{Y}_i - D_i H_i F_i \underline{\hat{X}}) \tag{2}$$

where the first term $||S\underline{\hat{X}}||_V^2$ penalizes for the nonsmoothness of the obtained estimation. This term is a regularization mechanism, which algebraically regularizes the problem to have a unique optimal solution, and physically forces the estimation result to satisfy our *a priori* knowledge about the smoothness of the ideal desired image. The matrix S is typically chosen to be the Laplacian operator, and the matrix V reflects our desire to apply locally adaptive smoothness, in order not to smooth edges.

The $[M_i^2 \times M_i^2]$ weight matrix W_i can be used to give different weighting for different measured images or, within an image, between different pixels, according to our confidence in the measurements. This confidence is dictated by the additive noise autocorrelation matrix—low variance noise translates to higher confidence.

Minimizing ϵ^2 with respect to $\underline{\hat{X}}$ is a classic least squares (LS) formulation, yielding the following set of L^2 linear equations:

$$\frac{\partial \epsilon^2}{\partial \underline{\hat{X}}} = 0 = -\left[S^T V S + \sum_{i=1}^{N} F_i^T H_i^T D_i^T W_i D_i H_i F_i \right] \underline{\hat{X}} + \sum_{i=1}^{N} F_i^T H_i^T D_i^T W_i \underline{Y}_i. \tag{3}$$

In order to solve the above set of linear equations, we have to invert a sparse $[L^2 \times L^2]$ matrix, which is practically impossible. Iterative methods, such as the steepest descent (SD), conjugate gradient (CG) and error relaxation methods such as the Jacoby, the Gauss–Seidel, and the successive overrelaxation (SOR), can be considered [16]. Using these algorithms, the convergence to the optimal solution is assured and is quite rapid (about 5–15 iterations are required [13]).

The above mentioned algorithms are adequate for tasks such as satellite fusion of several images and in combining several sources to improve targets detection and identification, or generating an improved still picture in VCR's. However, when applied to a typical video scene, the production of one superresolution output image cannot be regarded as a restoration of the scene. The required output should be a sequence of superresolution images with the same length as the source sequence and the same geometric behavior. The proposed new approach overcomes those limitations, presenting a time-space variant adaptive filter which produces an enhanced output sequence with the possibility of a superresolution feature.

III. Adaptive Filtering of Continuous Image Sequence

A. The Model and Performance Criterion

Let us start with a modification of our earlier notation to accommodate for the dependence on time. A measured sequence of images $\underline{Y}(t)$ is given where t is the discrete time index. We assume that there exists an ideal image-sequence of a fixed higher resolution, $\underline{X}(t)$, and that $\underline{Y}(t)$ is generated

from $\underline{X}(t)$ via

$$\underline{Y}(t) = DH(t)\underline{X}(t) + \underline{U}(t) \tag{4}$$

where D is the $[M^2 \times L^2]$ decimation matrix assumed here to be constant for all the measurements. The matrix $H(t)$ is the $[L^2 \times L^2]$ space and time varying blur of the tth high-resolution image, creating the tth measured low-resolution image. The dimensions of these matrices correspond to the resolution L^2 of the images $\underline{X}(t)$, and the resolution M^2 of the images $\underline{Y}(t)$, with the downsampling degradation effect. The vector $\underline{U}(t)$ is the measurement error in the model at time t, assumed to be random noise.

A second equation in the assumed model is an equation representing the correlation (or development) in time of the ideal image sequence, given by

$$\underline{X}(t-i) = F(t,i)\underline{X}(t) + \underline{S}(t,i) \tag{5}$$

where $F(t,i)$ is the $[L^2 \times L^2]$ matrix representing the geometric warp performed on $\underline{X}(t)$ yielding $\underline{X}(t-i)$, and $\underline{S}(t,i)$ is this model equation error, also assumed random noise. The above equation is deliberately presented in a noncausal form because of our aim to express an estimation of $\underline{X}(t)$ as a linear causal combination of the measured images $\underline{Y}$, as will be shown later.

Combining (4) and (5), a new model can be suggested, where each of the past N measured images can be related to the present image:

$$\begin{aligned}\underline{Y}(t-i) = &DH(t-i)F(t,i)\underline{X}(t) \\ &+ \underline{E}(t,i) \quad \text{for } 0 \le i \le N-1.\end{aligned} \tag{6}$$

$\underline{E}(t,i)$ is an overall model error between the measured image at $t-i$ and the model at time t, assumed random noise with autocorrelation matrix $W(t,i)^{-1}$. The motivation for this model comes from our attempt to treat the continuous image sequence restoration problem in a framework similar to the one adopted in the previous section. In this case, the N measured images $\underline{Y}(t-i)$, where $0 \le i \le N-1$, are presented as generated from a unique source image $\underline{X}(t)$, in a similar fashion to the relation given in (1). Applying the superresolution solution proposed before at time t on the last N measured images we get the following optimality criteria:

$$\begin{aligned}\epsilon(t)^2 &= \sum_{i=0}^{N-1} ||\underline{Y}(t-i) - DH(t-i)F(t,i)\hat{\underline{X}}(t)||^2_{W(t,i)} \\ &= \sum_{i=0}^{N-1} [\underline{Y}(t-i) - DH(t-i)F(t,i)\hat{\underline{X}}(t)]^T \\ &\quad W(t,i)[\underline{Y}(t,i) - DH(t-i)F(t,i)\hat{\underline{X}}(t)]\end{aligned} \tag{7}$$

which is similar to the criteria presented in (3) with time indices appropriately inserted (the only difference is the lack of the regularization term). $\hat{\underline{X}}(t)$ is the estimated version of $\underline{X}(t)$. The matrix $W(t,i)$ serves as a weighting matrix assumed symmetric and positive definite for all t and i. The optimal image $\hat{\underline{X}}(t)$, minimizing $\epsilon(t)^2$, is the solution of

$$\begin{aligned}\frac{\partial \epsilon(t)^2}{\partial \hat{\underline{X}}(t)} = 0 \Longrightarrow &\sum_{i=0}^{N-1} F(t,i)^T H(t-i)^T D^T W(t,i)\underline{Y}(t-i) \\ &= \left[\sum_{i=0}^{N-1} F(t,i)^T H(t-i)^T \right. \\ &\quad \left. D^T W(t,i)DH(t-i)F(t,i)\right]\hat{\underline{X}}(t)\end{aligned} \tag{8}$$

which can be written as

$$\Longrightarrow \underline{P}(t) = \boldsymbol{R}(t)\hat{\underline{X}}(t) \tag{9}$$

where

$$\underline{P}(t) = \sum_{i=0}^{N-1} F(t,i)^T H(t-i)^T D^T W(t,i)\underline{Y}(t-i) \tag{10}$$

and

$$\boldsymbol{R}(t) = \sum_{i=0}^{N-1} F(t,i)^T H(t-i)^T D^T W(t,i)DH(t-i)F(t,i). \tag{11}$$

Let us now assume a large data set, namely, $N \longrightarrow \infty$, and that $W(t,k) = \lambda^k \hat{W}$ (with $0 < \lambda < 1$). This results in an exponentially decaying weighting of past data. Then (10) and (11) become

$$\underline{P}(t) = \sum_{i=0}^{\infty} \lambda^k F(t,i)^T H(t-i)^T D^T \hat{W}\underline{Y}(t-i) \tag{12}$$

and

$$\boldsymbol{R} = \sum_{i=0}^{\infty} \lambda^k F(t,i)^T H(t-i)^T D^T \hat{W} DH(t-i)F(t,i). \tag{13}$$

Furthermore, by simple substitution in (5), it can be observed that $F(t,i)$ can be decomposed as follows:

$$\begin{aligned}F(t,i) &= F(t-1,i-1)F(t,1) = \cdots \\ &= F(t-i+1,1)F(t-i+2,1)F(t-i+3,1) \\ &\quad \cdots F(t-1,1)F(t,1) \\ &= \prod_{j=1}^{i} F(t-i+j,1)\end{aligned} \tag{14}$$

where we define $F(t,0) = I$ (Identity matrix). Equations (12) and (13) readily imply that

$$\begin{aligned}\underline{P}(t) &= \lambda F(t,1)^T \underline{P}(t-1) + H(t)^T D^T \hat{W}\underline{Y}(t) \\ \boldsymbol{R}(t) &= \lambda F(t,1)^T \boldsymbol{R}(t-1)F(t,1) + H(t)^T D^T \hat{W} DH(t)\end{aligned} \tag{15}$$

B. The Pseudo-RLS Algorithm

Using the model and criterion presented in (5), (6), and (15), we basically need to solve (9) at each time instant t. The solution of (9) is clearly given by

$$\hat{\underline{X}}(t) = \boldsymbol{R}(t)^{-1}\underline{P}(t). \tag{16}$$

However, the direct inversion of $\boldsymbol{R}(t)$ raises several difficulties which render this approach impractical in our case. The main one is the computational burden required here. In a typical video application the size of the matrix is $L^2 \times L^2$ (e.g.,

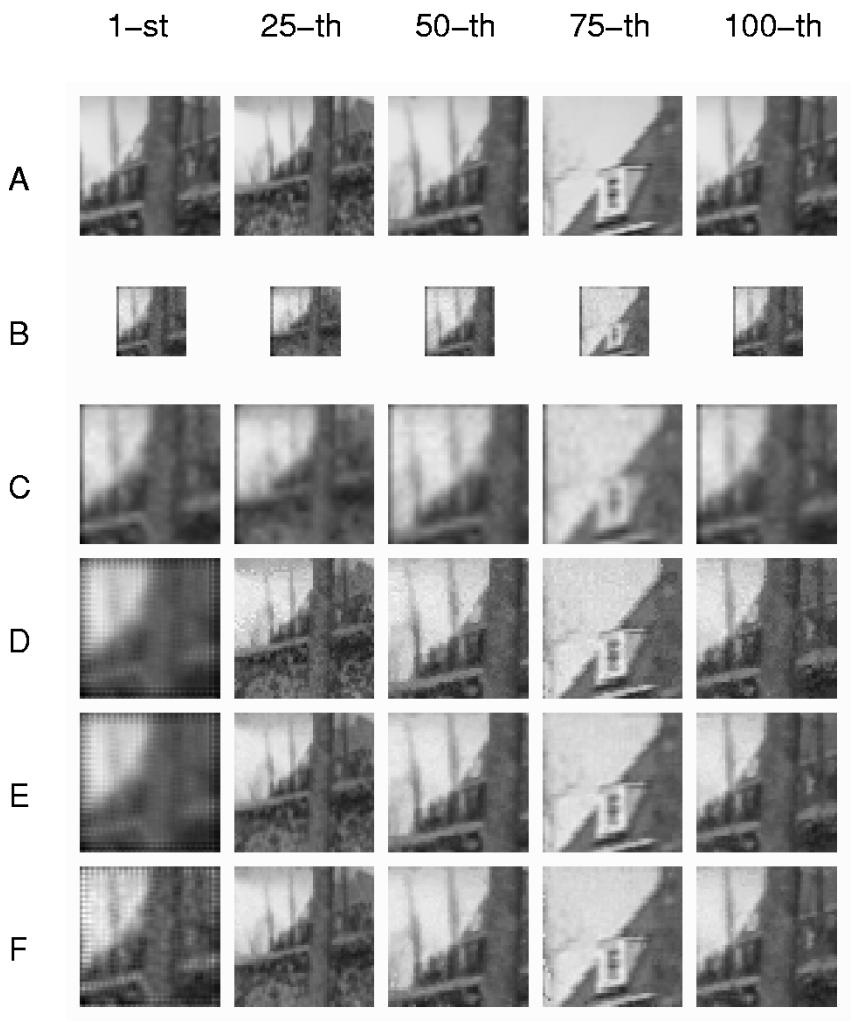


Fig. 1. First sequence results.

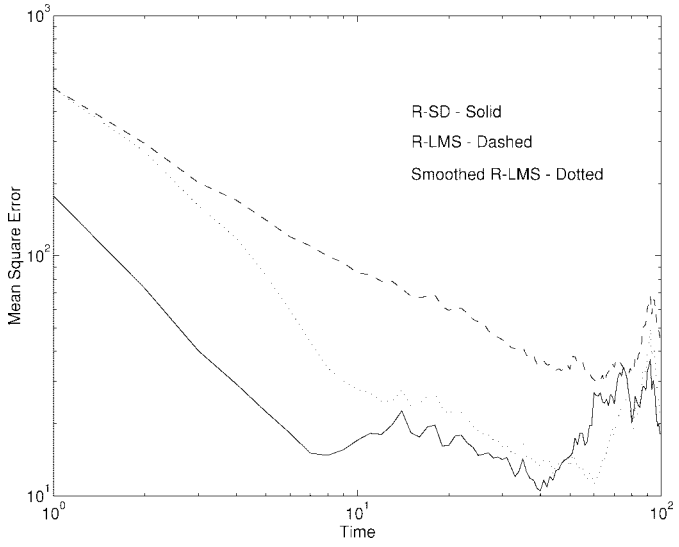


Fig. 2. First sequence—MSE versus time.

for $L = 1000$ we will be required to invert a $10^6 \times 10^6$ matrix). This problem also makes the RLS approach (see [14]), impractical here. Therefore, we need a computationally efficient way to calculate $\underline{\hat{X}}(t)$ at each time t. Since the criterion is LS, we named, generically, any algorithm that uses (14) to update $\underline{P}(t)$ and $\boldsymbol{R}(t)$ and then efficiently solves (9), pseudo-RLS algorithm. Namely, the pseudo-RLS algorithm involves two steps: The time update of the matrices $\boldsymbol{R}(t)$ and $\underline{P}(t)$ and the computation of the updated output image $\underline{\hat{X}}(t)$. The matrix $\boldsymbol{R}(t)$ can be approximated as a sparse matrix, using the fact that $\hat{H}(t)^T\hat{H}(t)$ is sparse band-limited (i.e., concentrated on a thin band around the diagonal), and the fact that the history matrix is weighted by λ, which exponentially decays its contribution at later time.

There has been a considerable effort in recent years at solving a linear sparse set of equations $A\underline{X} = \underline{B}$ (see, e.g., [16]–[18]). The proposed methods are divided into two major classes—iterative and direct. As mentioned earlier, in this work we have chosen to concentrate on the iterative methods because of their relative simplicity, their ability to work with various sparseness forms, their relatively acceptable performance, and most important—their property of not requiring the inverse matrix, which considerably reduces computer memory requirements.

Among the iterative algorithms, the SD, the normalized SD (NSD), the SOR, the CG, and the iterated defect correction (IDC) can be considered as good candidates [16]–[18], which comprise simplicity and performance.

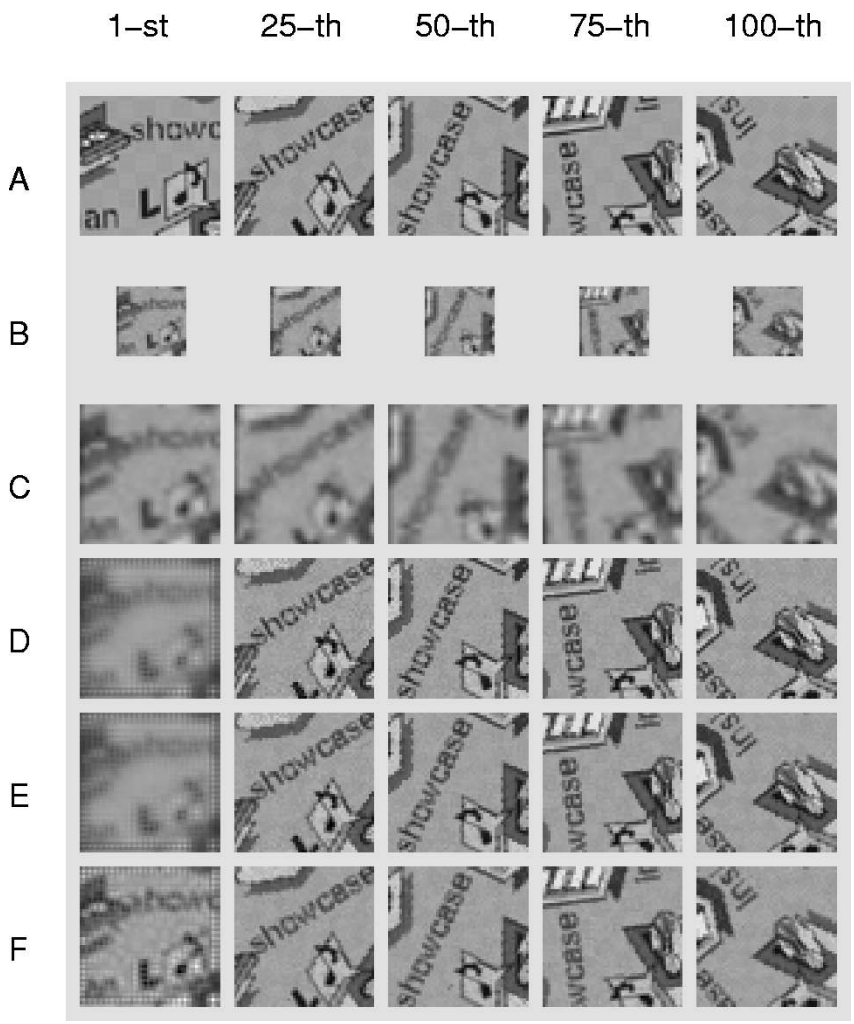


Fig. 3. Second sequence results.

In particular, we have used both SD and NSD to iteratively solve (9). These are given by

$$\underline{\hat{X}}_{j+1}(t) = \underline{\hat{X}}_j(t) + \mu_j(t)[\underline{P}(t) - \boldsymbol{R}(t)\underline{\hat{X}}_j(t)] \tag{17}$$

where $\mu_j(t) = \mu$ (constant) for the SD, and

$$\mu_j(t) = \frac{\underline{Z}_j(t)^T \underline{Z}_j(t)}{\underline{Z}_j(t)^T \boldsymbol{R}(t) \underline{Z}_j(t)} \tag{18}$$

for the NSD, where $\underline{Z}_j(t) = \underline{P}(t) - \boldsymbol{R}(t)\underline{\hat{X}}_j(t)$. At each time t we perform K iterations and we initialize at time t using the previous temporal point solution. Namely, $\underline{\hat{X}}_0(t) = G(t,1)\underline{\hat{X}}_K(t-1)$, where $G(t,1)$ is the pseudo-inverse of $F(t,1)$, representing the forward warp operator.

Remark: Another problem with solving (9) may arise with $\boldsymbol{R}(t)$ being singular or with large condition number. This may happen even if $\boldsymbol{R}(t-1)$ is nonsingular. To alleviate this problem a regularization term can be added (see [13]) to the LS criterion in equation (7) which, in turn, results in an additional term in the equation for $\boldsymbol{R}(t)$ guaranteeing its nonsingularity. A commonly used regularization term is of the form $\beta \|S\underline{\hat{X}}(t)\|^2_{V(t)}$, similar to the form used in (2) in the previous section. Then the upgrading equation for $\boldsymbol{R}(t)$ becomes

$$\begin{aligned}\boldsymbol{R}(t) = {} & \lambda F(t,1)^T \boldsymbol{R}(t-1) F(t,1) \\ & + \beta S^T S + H(t)^T D^T \hat{W} D H(t)\end{aligned} \tag{19}$$

and $\boldsymbol{R}(t)$ is guaranteed to be nonsingular.

C. The LMS Algorithm

Further simplification can be achieved by using the type of approximation used in the literature to get the LMS algorithm from the SD algorithm. Namely, if we replace in (17) $\underline{P}(t)$ by $H(t)^T D^T \hat{W} \underline{Y}(t)$ and $\boldsymbol{R}(t)$ by $H(t)^T D^T \hat{W} D H(t)$, we get the LMS algorithm

$$\underline{\hat{X}}_{j+1}(t) = \underline{\hat{X}}_j(t) + \mu H(t)^T D^T \hat{W}[\underline{Y}(t) - DH(t)\underline{\hat{X}}_j(t)] \tag{20}$$

where again, we use K such iterations.The initial condition at each time t can be chosen in a way similar to the way it is presented for the pseudo-RLS, namely, $\underline{\hat{X}}_0 = G(t,1)\underline{\hat{X}}_K(t-1)$.

Note that the above approximation results when, in (14), $\lambda = 0$. Then the optimality criterion becomes the instantaneous squared error and the algorithm, at each time instant

t, uses only present data to modify the projected image $G(t,1)\underline{\hat{X}}_K(t-1)$.

We have commented earlier, for the pseudo-RLS algorithm, that adding a regularization term can improve the algorithm performance. This is applicable here as well. In that case, the instantaneous squared error contains a second term penalizing for norm or smoothness features according to requirements.

Initialization of the algorithm can be done by assuming zero value image at $t = 0$, or by choosing an interpolated version of $\underline{Y}(1)$, the first data image. New information in the scene raises a direct extension of the initialization problem. Such information correspond to boundaries entering the scene or hidden objects emerging from occlusion. Since the LMS algorithm basically performs smoothing in the time axis to decay noise, and since no history information is given on the new data, the output value which corresponds to the new data must be initialized. The recommended approach in such cases is to use the low-resolution measure data as the initial guess, interpolated and smoothed adequately.

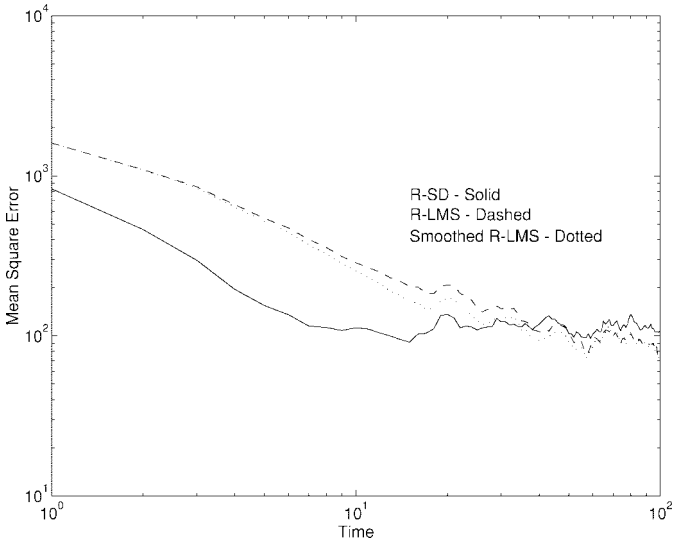


Fig. 4. Second sequence—MSE versus time.

IV. Computational Complexity

A. General

As was explained in the previous section, the estimation of the restored superresolution sequence comes with sever computational burden. This is a direct consequence of having very large dimensional vectors and matrices.

The LMS algorithms are computationally simpler, compared to the pseudo-RLS. Whereas the pseudo-RLS requires the update of $\boldsymbol{R}(t)$ and $\underline{P}(t)$ at each temporal point, the LMS does not update, nor stores these values. Beyond this part, the two algorithms are similar if the same number of iterations is applied at each temporal point.

Throughout the following computation complexity analysis, for simplicity, only multiplications are considered. We further assume the following.

1) The applied algorithm is the SD algorithm.
2) The measurement noise autocorrelation matrix is $\hat{W} = I$.
3) The blur matrix $H(t)$ represents a general spatial-temporal dependent blur kernel of size p^2.
4) The warp matrices $G(t,1)$ and $F(t,1)$ are nearest neighbor warp matrices. As such, each line in these matrices contains only one nonzero entry equal to one. Thus, there are no multiplications involved in applying the warp on a given vector.
5) The regularization term is omitted.
6) The parameter μ is a predetermined constant.

B. Iterative Solution in the Pseudo-RLS Algorithm

The pseudo-RLS algorithm at time t consists of three major steps: 1) Warping the previous output in order to create the initial solution—no multiplications are involved in this step; 2) Updating the pair $\boldsymbol{R}(t)$ and $\underline{P}(t)$; and 3) Performing the K iterations of the SD algorithm. The update of the pair $\boldsymbol{R}(t)$ and $\underline{P}(t)$ requires the following steps.

1) The multiplication $DH(t)$ require no multiplications, since all the entries in D are one. Each column of this matrix contains p^2M^2/L^2 nonzero entries.
2) The computation of $H(t)^TD^TDH(t)$ requires less than p^4M^4/L^2 multiplications.
3) The multiplication $H(t)^TD^T\underline{Y}(t)$ requires p^2M^2 multiplications.
4) The multiplications $F(t,1)^T\underline{P}(t-1)$ and $F(t,1)^T\boldsymbol{R}(t-1)F(t,1)$ require no multiplications because of the content of $F(t,1)$.
5) Multiplying the above two terms with λ requires $L^2[L^2d(t-1)+1]$ multiplications, where $d(t-1)$ is the density of the matrix $\boldsymbol{R}(t-1)$, defined as the number of nonzero entries in $\boldsymbol{R}(t-1)$ divided by the total number of the matrix entries.

The application of one iteration of the SD with the updated pair $\boldsymbol{R}(t)$ and $\underline{P}(t)$ requires

1) the multiplication of $\boldsymbol{R}(t)\underline{\hat{X}}_j(t)$, which requires $L^4d(t-1)$ multiplications;
2) the multiplication of $\underline{P}(t) - \boldsymbol{R}(t)\underline{\hat{X}}_j(t)$ by μ which requires L^2 multiplications.

The overall number of multiplications per K iterations and per one output image is therefore

$$\begin{aligned} C_{P-RPS} &= L^2[L^2d(t-1)+1](K+1) \\ &\quad + M^2p^2\left(1+\frac{p^2M^2}{L^2}\right) \quad \frac{\text{Multiplications}}{\text{output image}} \\ &= [L^2d(t-1)+1](K+1) \\ &\quad + \frac{M^2p^2}{L^2}\left(1+\frac{p^2M^2}{L^2}\right) \quad \frac{\text{Multiplications}}{\text{output pixel}}. \end{aligned} \tag{21}$$

For example, for the case where $M = 500$, $L = 1000$, $p = 5$, and $K = 5$ with $d(t-1) = 1e-4$ (typical value taken from the simulations), it is required to perform approximately 650 multiplications per one output pixel. This is roughly equivalent to the application of 8×8 blur kernel ten times.

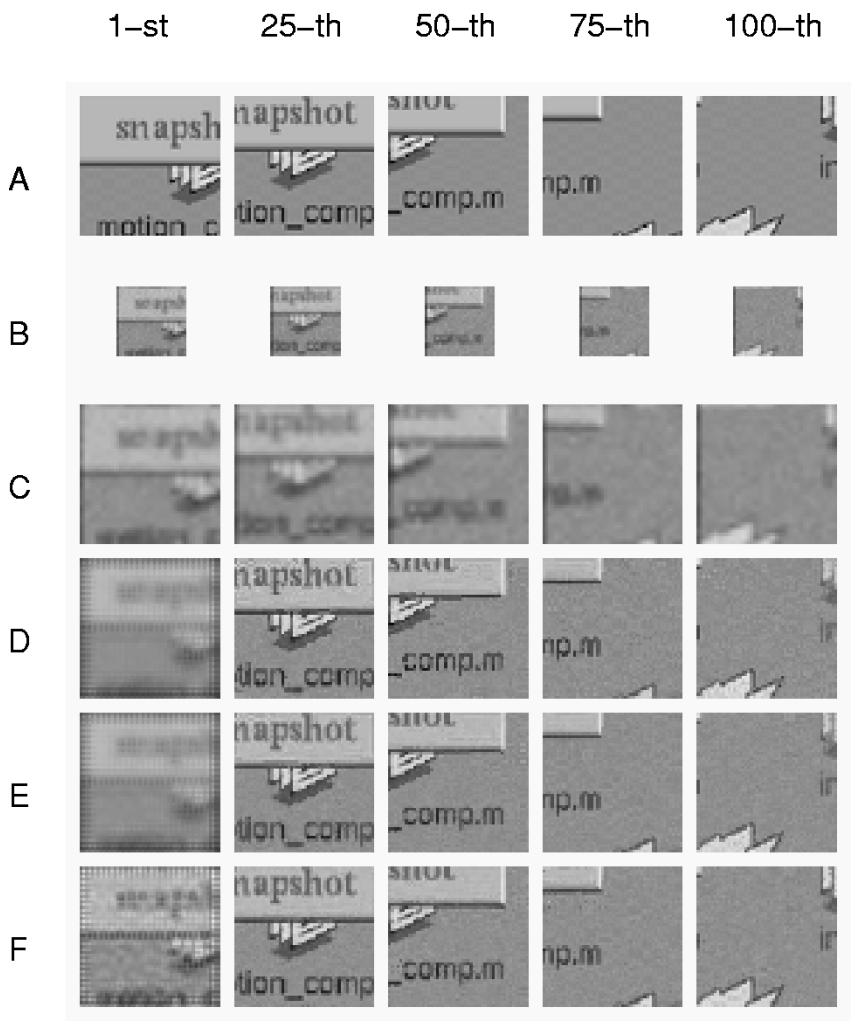


Fig. 5. Third sequence results.

C. The LMS Algorithm

We refer to equation (20) as the set of the operations to be applied within the LMS algorithm. Each iteration can be broken into several parts according to the following.

1) The term $DH(t)\underline{\hat{X}}_j(t)$ requires M^2p^2 multiplications.
2) The term $H(t)^T D^T[\underline{Y}(t) - DH(t)\underline{\hat{X}}_j(t)]$ requires M^2p^2 more multiplications.
3) Multiplying by μ requires M^2 multiplications (by performing it on the vector $\underline{Y}(t) - DH(t)\underline{\hat{X}}_j(t)$.

The overall number of multiplications per K iterations and per one output image is therefore

$$C_{LMS} = K(2p^2+1)M^2 \frac{\text{Multiplications}}{\text{output image}} = K(2p^2+1)\frac{M^2}{L^2} \quad \frac{\text{Multiplications}}{\text{output pixel}} \tag{22}$$

For example, for the case where $M = 500, L = 1000, p = 5$, and $K = 5$, it is required to perform approximately 63.75 multiplications per one output pixel. This is roughly equivalent to the application of 8×8 blur kernel, which is definitely accepted as applicable in real time.

In the case where a two-dimensional (2-D) filter is time and space invariant blur function, the structure of its matrix representation is a generalized block Toeplitz [1]. This structure can be exploited to further reduce the required computations for the implementation of this filter. More details on such computational savings in this case can be found in [19].

V. Simulation Results

In this section, we present three examples, through which we demonstrate the feasibility of the proposed approach for superresolution image sequence restoration. In these examples we demonstrate the performance of the LMS (with and without regularization) and the pseudo-RLS (with regularization) methods, as were presented in Section III.

The three tests are based on synthetic image sequences, each containing 100 images of size [50 × 50] pixels. These sequences serve as the ideal images. The measured image sequences were generated from the above sequences by blurring each image using [3 × 3] uniform kernel, decimation of ratio 2 : 1, and adding Gaussian zero mean white noise with $\sigma = 5$ (the dynamic range of the gray level is 0 to 255). Thus, the measured sequences contain 100 images of size [25 × 25] pixels each. These dimensions were chosen in order to shorten the simulations run-time and because of the memory limitations of MATLAB.

The pseudo-RLS algorithm was implemented using the NSD iterative algorithm, using ten iterations per each time point. In all cases, the initialization at $t = 0$ was taken to be a bilinear interpolated version of the first measured image. Initialization of new information in the scene (i.e., boundary problem) was taken to be zeros. The regularization in the above tests was the Laplacian, with regularization parameter $\beta = 0.02$. In the restoration process, the actual blur, decimation, and motion matrices were used. Incorporation of motion estimation into the restoration process requires the use of motion estimation algorithms, which are beyond the scope of this paper. This is the reason we used synthetic sequences to begin with. The LMS was used with only one iteration at each time t.

In Fig. 1, the results of the first test are presented. In order to illustrate the temporal axis, the first, 25th, 50th, 75th, and 100th images of each sequence are given. The motion in this sequence consist of global translation motion. The given sequences are: A—ideal sequence; B—measured sequence; C—bilinear interpolation of the measurements; D—LMS results; E— regularized LMS results; F—regularized pseudo-RLS results.

Fig. 2 presents the MSE between the restored and the ideal sequences for each method in the above test. The MSE is presented using logarithmic time scale in order to give an expanded view of the initialization process.

Similar to Fig. 1, Figs. 3 and 5 present the results for the second and third sequences. The motion in the second sequence consists of global zoom in and out and global rotation motion. The motion in the third sequence consists of global translational motion. Figs. 4 and 6 give the restoration MSE for these tests.

From the figures, we can observe the following.

1) In all the proposed restoration methods in all examples there is a clear improvement both from the resolution quality and the suppression of the degradation effects.
2) The LMS method without regularization is slightly weaker, compared to the regularized LMS and pseudo-RLS methods. This is true both for the convergence rate at the initialization period and the steady state error.
3) As expected, the pseudo-RLS method has a shorter initialization period, compared to the LMS algorithm.

VI. Summary and Conclusion

A new general linear model has been presented here for the problem of superresolution restoration of continuous image sequence. This model is a generalization of the model used by the classic superresolution problem as presented in [13]. Based on this model and the adaptive filtering theory, several algorithms have been proposed, enabling the restoration of a superresolution image sequence from a low-resolution space- and time-variant blurred image sequence, assuming the existence of additive white noise and arbitrary motion. These algorithms are based on the RLS and the LMS methods, allowing for the incorporation of prior knowledge about the solution into the restoration process as a regularization term.

The approach presented in this work was shown to be applicable with an order of L^2 multiplications per image, in various forms, where L^2 is the number of pixels in one output image. The LMS and pseudo-RLS algorithms were tested in extensive simulations, giving encouraging results. A sample of these results is presented here.

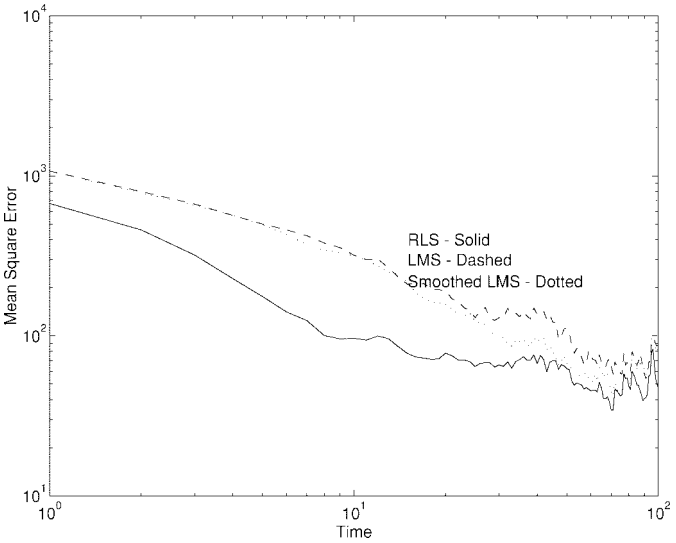


Fig. 6. Third sequence—MSE versus time.

There are several important open issues related to the presented algorithms:

1) the convergence properties of the LMS algorithm for the treated problem;
2) approximation of the matrix $\boldsymbol{R}(t)$ in the pseudo-RLS algorithm as a sparse matrix for all t;
3) the need to estimate the motion and analyze the effects of its errors on the proposed restoration quality.

These questions and others are currently under investigation.

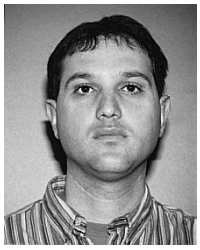
Acknowledgment

The authors are grateful to Prof. A. Brukstein and Dr. G. Shapiro for their useful suggestions and illuminating remarks, and to R. Kimmel and D. Shaked for fruitful and helpful discussions.

References

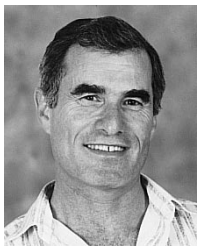
[1] W. K. Pratt, *Digital Image Processing*, 2nd ed. New York: Wiley, 1991.
[2] T. S. Huang, *Image Sequence Analysis*, 1st ed. Berlin, Germany: Springer-Verlag, 1981.
[3] ——, *Image Sequence Processing and Dynamic Scene Analysis*, 2nd ed. Berlin, Germany: Springer-Verlag, 1983.
[4] A. J. Patti and A. M. T. M. I. Sezan, "Image sequence restoration and de-interlacing by motion compensated Kalman filtering," *Proc. SPIE*, vol. 1903, pp. 59–70, 1991.
[5] S. J. Ko and Y. H. Lee, "Nonlinear spatio-temporal noise suppression techniques with applications in image sequence processing," in *IEEE Int. Symp. Circuits Syst.*, vol. 5, pp. 662–665, 1991.
[6] A. K. Katsaggelos, R. P. Kleihorst, S. N. Efstratiadis, and R. L. Lagendijk, "Adaptive image sequence noise filtering methods," in *Proc. SPIE*, vol. 1606, pp. 716–727, 1991.
[7] S. P. Kim, N. K. Bose, and H. M. Valenzuela, "Recursive reconstruction of high resolution image from noisy undersampled multiframes," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, pp. 1013–1027, June 1990.

[8] C. Srinivas and M. D. Srinath, "A stochastic model-based approach for simultaneous restoration of multiple misregistered images," in *Proc. SPIE*, vol. 1360, pp. 1416–1427, 1990.
[9] H. Ur and D. Gross, "Improved resolution from subpixel shifted pictures," *CVGIP: Graph. Models Image Process.*, vol. 54, pp. 181–186, Mar. 1992.
[10] M. Irani and S. Peleg, "Motion analysis for image enhancement: Resolution, occlusion, and transparency," *J. Vis. Commun. Image Represent.*, vol. 4, pp. 324–335, Dec. 1993.
[11] A. J. Patti and A. Tekalp, "High resolution image reconstruction from a low-resolution image sequence in the presence of time-varying motion blur," in *Proc. ICIP*, Austin, TX, 1994, pp. 343–347.
[12] R. Schultz and R. Stevenson, "Extraction of high-resolution frames from video sequences," *IEEE Trans. Image Processing*, vol. 5, pp. 996–1011, June 1996.
[13] M. Elad and A. Feuer, "Restoration of a single super-resolution image from several blurred, noisy and under-sampled measured images," *IEEE Trans. Image Processing*, vol. 6, pp. 1646–1658, Dec. 1997.
[14] S. Haykin, *Adaptive Filter Theory*, 1st ed. Englewood Cliffs, NJ: Prentice-Hall, Inc., 1986.
[15] C. K. Chui and G. Chen, *Kalman Filtering*, 2nd ed. Berlin, Germany: Springer-Verlag, 1990.
[16] L. A. Hageman and D. Young, *Applied Iterative Methods*, 1st ed. New York: Academic, 1981.
[17] A. George, *Computer Solution of Large Sparse Positive Definite Systems*, 1st ed.. Englewood Cliffs, NJ: Prentice-Hall, 1981.
[18] I. S. Duff, *Sparse Matrices and Their Uses*, 1st ed. New York: Academic, 1981.
[19] M. Elad and A. Feuer, Tech. Rep. 973, Technion—Israel Inst. Technol., Haifa, 1994.



Michael Elad (M'98) was born in Haifa, Israel, in December 1963. He received the B.Sc., M.Sc., and Ph.D. degrees from the Electrical Engineering Department, Technion—Israel Institute of Technology, Haifa, in 1986, 1988, and 1997, respectively.

Since January 1997, he has been with Hewlett-Packard Laboratories, Technion City, Haifa, as a Research and Development Engineer. His current research interests include image reconstruction problems, adaptive filtering theory applied to image processing, and optimization theory in image processing, computer vision, and pattern recognition.



Arie Feuer (S'74–M'86–SM'93) received the B.Sc. and M.Sc. degrees from the Technion—Israel Institute of Technology, Haifa, in 1967 and 1973, respectively, and the Ph.D. degree from Yale University, New Haven, CT, in 1978.

From 1967 to 1970, he was with Technomatic Israel, working on factory automation. From 1978 through 1983, he was with Bell Laboratories, Holmdel, NJ, studying telephone networks performance. Since 1983, he has been with the Electrical Engineering Department, Technion. His research interests are in adaptive systems and sampled data systems, both in control and in signal and image processing.