A TRUE ORDER RECURSIVE ALGORITHM FOR TWO-DIMENSIONAL LEAST SQUARES ERROR LINEAR PREDICTION AND FILTERING

George-Othon Glentis
TEI of Heraklion, Branch at Chania, Department of Electronics
3, Romanou Str, Halepa, Chania 73133, Greece

ABSTRACT

In this paper a novel algorithm is presented for the efficient Two-Dimensional (2-D), Least Squares (LS) FIR filtering and system identification. Causal filter masks of general boundaries are allowed. Efficient order updating recursions are developed by exploiting the spatial shift invariance property of the 2-D data set. Single step order updating recursions are developed. During each iteration, the filter coefficients set is augmented by a single new element. The single step order updating formulas allow for the development of an efficient, true order recursive algorithm for the 2-D LS causal linear prediction and filtering.

1 Introduction

Two Dimensional Least Squares filtering and system identification are of great importance in a wide range of applications. These include image restoration, image enhancement, image compression, 2-D spectral estimation, detection of changes in image sequences, stochastic texture modeling, edge detection etc, [1],[2],

Let $x(n_1,n_2)$ be the input of a linear, space invariant, 2-D FIR filter. The filter's output $y(n_1,n_2)$ is a linear combination of past input values $x(n_1-i_1,n_2-i_2)$ weighted by the *filter coefficients* c_{i_1,i_2} over a support region, or *filter mask*, $\mathcal{M}$

$$y(n_1,n_2) = -\sum_{(i_1,i_2)\in\mathcal{M}} c_{i_1,i_2}x(n_1-i_1,n_2-i_2) \qquad (1)$$

Two-Dimensional support regions of general causal (first quarter) shapes are considered. Noncausal support regions can be handled in a similar way. Let $\mathcal{M}$ can be causal. We write $\mathcal{M}$ as a union of horizontal strips, as

$$\mathcal{M} = \cup_{i_1=0}^{l_1}\mathbf{m}(i_1)$$
$$\mathbf{m}(i_1) = \{(i_1,i_2) : 0 \le i_2 \le l_2(i_1)\}$$

where, $l_1 = \max\{i_1 : (i_1,i_2) \in \mathcal{M}\}$, and $l_2(i_1) = \max\{i_2 : (i_1,i_2) \in \mathbf{m}(i_1)\}$.

Let us define the data vector $\mathbf{x}_{m(i_1)}(n_1,n_2)$, for all $i_1 \in [0,l_1]$, which consists of all data laying on the i_1-th row, i.e., $\mathbf{m}(i_1)$, of the filter mask $\mathcal{M}$

$$\mathbf{x}_{m(i_1)}(n_1,n_2) = [\ x(n_1-i_1,n_2)\ x(n_1-i_1,n_2-1)\ \ldots\ x(n_1-i_1,n_2-l_2(i_1)+1)\ \ x(n_1-i_1,n_2-l_2(i_1))\ ]^t$$

Superscript ^{t} means transpose. In a similar way, define the coefficients vector corresponding to $\mathbf{m}(i_1)$

$$\mathbf{c}_{m(i_1)} = [c_{i_1,0}\ c_{i_1,+1}\ c_{i_1,k_2(i_1)+2}\ \cdots c_{i_1,l_2(i_1)-1}\ c_{i_1,l_2(i_1)}]^t$$

Then, the data vector and the coefficients vector corresponding to the mask $\mathcal{M}$, take the form

$$\begin{aligned}\boldsymbol{\mathcal{X}}_{\mathcal{M}}(n_1,n_2) &= [\ \mathbf{x}^t_{m(0)}(n_1,n_2)\ \mathbf{x}^t_{m(1)}(n_1,n_2)\ \ldots \\ &\quad \mathbf{x}^t_{m(l_1-1)}(n_1,n_2)\ \mathbf{x}^t_{m(l_1)}(n_1,n_2)\ ]^t \\ \boldsymbol{\mathcal{C}}_{\mathcal{M}} &= [\mathbf{c}^t_{m(0)}\ \mathbf{c}^t_{m(1)}\ \mathbf{c}^t_{m(k_1+2)}\ \cdots \mathbf{c}^t_{m(l_1-1)}\ \mathbf{c}^t_{m(l_1)}\ ]^t\end{aligned} \qquad (2)$$

Using definitions (2), eq. (1) can be written in a compact, linear regression form, [6],

$$y(n_1,n_2) = -\boldsymbol{\mathcal{X}}^t_{\mathcal{M}}(n_1,n_2)\boldsymbol{\mathcal{C}}_{\mathcal{M}} \qquad (3)$$

The LS 2-D FIR filtering is casted as follows. Given a 2-D sequence of an input signal, $x(n_1,n_2)$, and a 2-D sequence of a desired response signal $z(n_1,n_2)$, over the rectangular data support region $\mathcal{S}$, estimate optimum coefficients of model (3), that minimize the cost function

$$\mathcal{E}_{\mathcal{M}}(N_1,N_2) = \sum_{n_1=0}^{N_1}\sum_{n_2=0}^{N_2}\left(z(n_1,n_2)-y(n_1,n_2)\right)^2 \qquad (4)$$

A prewindowing assumption has been adopted. The normal equations resulting from the minimization of the above cost function, are taken the form

$$\boldsymbol{\mathcal{R}}_{\mathcal{M}}(N_1,N_2)\boldsymbol{\mathcal{C}}_{\mathcal{M}}(N_1,N_2) = -\boldsymbol{\mathcal{D}}_{\mathcal{M}}(N_1,N_2) \qquad (5)$$

where

$$\boldsymbol{\mathcal{R}}_{\mathcal{M}}(N_1,N_2) = \sum_{n_1=0}^{N_1}\sum_{n_2=0}^{N_2}\boldsymbol{\mathcal{X}}_{\mathcal{M}}(n_1,n_2)\boldsymbol{\mathcal{X}}^t_{\mathcal{M}}(n_1,n_2) \qquad (6)$$

and

$$\boldsymbol{\mathcal{D}}_{\mathcal{M}}(N_1,N_2) = \sum_{n_1=0}^{N_1}\sum_{n_2=0}^{N_2}\boldsymbol{\mathcal{X}}_{\mathcal{M}}(n_1,n_2)z(n_1,n_2) \qquad (7)$$

$\mathcal{R}_{\mathcal{M}}(N_1, N_2)$ and $\mathcal{D}_{\mathcal{M}}(N_1, N_2)$ are the input signal sampled autocorrelation matrix, and the sampled cross correlation vector between the input and the desired response signal, respectively.

Any well behaved linear system solver can be applied for the inversion of the 2-D normal equations, (5). However, the special structure of the normal equations gives rise to the development of cost effective algorithms for the determination of the unknown parameters. The multichannel Levinson-Wiggins-Robinson (LWR) algorithm, [1], [8] is a well known example. A major feature these algorithms offer against the conventional counterparts, like Cholesky's method, is reduction of computational complexity by an order of magnitude.

The application of the multichannel LWR algorithm for the solution of the normal equations requires a columnwise (or a rowwise) organization of the filter mask. In this way, spatial shift invariance characteristics can be utilized. The normal equations take a highly structured near-to Toeplitz-block-Toeplitz form, [8]. The column(row)-wise approach, however, implies a severe restriction to system modeling since 2-D masks of rectangular shape can only be handed.

In this paper a fast algorithm is developed for the solution of the near-to Toeplitz-block-Toeplitz normal equations in an order recursive way. Filter masks of general shape are allowed. The prosposed algorithm extends the algorithms proposed in [4]-[7], that deal with the Mean Squared Error case, to the Least Squares counterpart. Efficient recursions are developed for updating of lower order filter parameters towards any neighboring point. It can be efficiently applied for the order recursive estimation of the 2-D Least Squares causal FIR filter and system identification, accelerating the exhaustive search procedures required by most of the order determination criteria, [9]-[10].

2 THE PROPOSED ALGORITHM

In this section order updating recursions are developed for the solution of the normal equations (5) for the prewindowing LS 2D filtering. An order recursive algorithm is proposed that serves for the transmission from lower order parameters to increased order counterparts. Single step increments of the filter mask $\mathcal{M}$ are allowed each time. The method however can be applied to the general case of noncausal support regions, and with some extra algebraic manipulation to the more general unwindowed LS case.

Thus, starting from $\mathcal{M}$ an increased order mask is constructed with one additional neighboring sample. Let us consider the i_1-th row of the filter mask, i.e., $\mathbf{m}(i_1), i_1 \in [0, l_1]$. It corresponds to $p(i_1) = l_2(i_1) + 1$ filter taps. The filter mask can be augmented by adding an an extra filter coefficient at $(i_1, l_2(i_1)+1)$, i.e., at the i_1-th row.

Let $\mathcal{M} + L(i_1)$ be the increased order filter mask. Thus,

$$\mathcal{M} + L(i_1) = \mathcal{M} \cup \{(i_1, l_2(i_1) + 1)\} \tag{8}$$

The corresponding augmented data vector (2) is partitioned as

$$\mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2) = \mathcal{S}^t_{L(i_1)} \begin{bmatrix} \mathcal{X}_{\mathcal{M}}(n_1, n_2) \\ x(n_1 - i_1, n_2 - l_2(i_1) - 1) \end{bmatrix} \tag{9}$$

$\mathcal{S}_{L(i_1)}$ is a permutation matrix utilized to extract (push down) the extra data sample $x(n_1 - i_1, n_2 - l_2(i_1) - 1)$ out of $\mathcal{X}_{\mathcal{M}+L(i_1)}(n_1, n_2)$.

Based on the data partition strategy for the data vectors associated with the increased order filter masks, efficient recursions are developed for updating of filter parameters $\mathcal{C}_{\mathcal{M}} \rightarrow \mathcal{C}_{\mathcal{M}+L(i_1)}$.

2.1 Filter order updating recursions

Consider the increased order linear system (5) corresponding to the augmented mask $\mathcal{M} + L(i_1)$. Then, it can be partitioned using (9) as

$$\mathcal{S}^t_{L(i_1)} \begin{bmatrix} \mathcal{R}_{\mathcal{M}}(N_1, N_2) & \mathbf{r}^{b(i_1)}_{\mathcal{M}}(N_1, N_2) \\ \mathbf{r}^{b(i_1)t}_{\mathcal{M}}(N_1, N_2) & \rho^{b(i_1)}_{\mathcal{M}}(N_1, N_2) \end{bmatrix} .$$
$$.\mathcal{S}_{L(i_1)}\mathcal{C}_{\mathcal{M}+L(i_1)}(N_1, N_2) = -\mathcal{S}^t_{L(i_1)} \begin{bmatrix} \mathcal{D}_{\mathcal{M}}(N_1, N_2) \\ d_{i_1, l_2(i_1)+1}(N_1, N_2) \end{bmatrix} \tag{10}$$

$$d(i_1, l_2(i_1) + 1)(N_1, N_2) =$$
$$\sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} x(n_1 - i_1, n_2 - l_2(i_1) - 1) z(n_1, n_2)$$
$$\mathbf{r}^{b(i_1)}_{\mathcal{M}}(N_1, N_2) =$$
$$\sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \mathcal{X}_{\mathcal{M}}(n_1, n_2) \ .x(n_1 - i_1, n_2 - l_2(i_1) - 1)$$

Application of the matrix inversion lemma for partitioned matrices, leads to a recursive way for the left hand side order update of $\mathcal{C}_{\mathcal{M}}(N_1, N_2)$, (see Table 1).

Auxiliary parameters $\mathbf{b}^{i_1}_{\mathcal{M}}(N_1, N_2)$ and $\mathbf{a}^{i_1}_{\mathcal{M}}(N_1, N_2)$ introduced in Table 1 can both be interpreted as single step backward and forward 2-D predictors. The development of an order recursive algorithm for the determination of the optimum filters $\mathcal{C}_{\mathcal{M}+L(i_1)}(N_1, N_2)$ for all possible augmented masks $\{(i_1, l_2(i_1) + 1)\}$ for all $i_1 \in [0, l_1]$, requires recursions for updating the backward predictors for all $\ell = 0 \ldots l_1$. The backward and forward predictors are obtained by setting the desired response signal $z(n_1, n_2) = x(n_1 - \ell, n_2 - l_2(\ell) - 1)$ and $z(n_1, n_2) = x(n_1 - \ell, n_2)$, respectively. They are estimated as the solutions of the normal equations

$$\begin{aligned} \mathcal{R}_{\mathcal{M}}(N_1, N_2)\mathbf{b}^{\ell}_{\mathcal{M}}(N_1, N_2) &= -\mathbf{r}^{b(\ell)}_{\mathcal{M}}(N_1, N_2) \\ \mathcal{R}_{\mathcal{M}}(N_1, N_2 - 1)\mathbf{a}^{\ell}_{\mathcal{M}}(N_1, N_2) &= -\mathbf{r}^{f(\ell)}_{\mathcal{M}}(N_1, N_2) \end{aligned} \tag{11}$$

for all $\ell \in [0, l_1]$. The backward and forward autocorrelation vectors are defined as

$$\mathbf{r}^{b(\ell)}_{\mathcal{M}}(N_1, N_2) =$$
$$\sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \mathcal{X}_{\mathcal{M}}(n_1, n_2) x(n_1 - \ell, n_2 - l_2(\ell) - 1)$$
$$\mathbf{r}^{f(\ell)}_{\mathcal{M}}(N_1, N_2) =$$
$$\sum_{n_1=0}^{N_1} \sum_{n_2=0}^{N_2} \mathcal{X}_{\mathcal{M}}(n_1, n_2 - 1) x(n_1 - i_1, n_2)$$

To be able to develop an order recursive algorithm for the determination of the optimum filter $\mathcal{C}_{\mathcal{M}}(N_1, N_2)$, recursions for updating the single channel backward and forward predictors are required. The derivation of recursions (5)-(7) and (15)-(24) of Table 1 follows [5] and [6]. The main difference is the space shift introduced at the backward predictor, eqs. (19) and (24). Notice that

$$\mathcal{R}_{\mathcal{M}}(N_1, N_2) = \mathcal{R}_{\mathcal{M}}(N_1, N_2 - 1) + \sum_{n_1=0}^{N_1} \boldsymbol{\mathcal{X}}_{\mathcal{M}}(n_1, N_2)\boldsymbol{\mathcal{X}}^t_{\mathcal{M}}(n_1, N_2)$$

and

$$\mathbf{r}^{b(i_1}_{\mathcal{M}})(N_1, N_2) = \mathbf{r}^{b(i_1}_{\mathcal{M}})(N_1, N_2 - 1) + \sum_{n_1=0}^{N_1} \boldsymbol{\mathcal{X}}_{\mathcal{M}}(n_1, N_2)x(n_1 - i_1, n_2 - l_2(i_1))$$

The extra recursions are required to compensate for the space shift, eqs. (8)-(14) of Table 1.

2.2 Overall organization

The order recursive equations developed so far, for the updating of the filter coefficients vector, as well as for the auxiliary backward and forward single step predictors, can be tight together to form a powerful **true** order recursive 2-D algorithm. Indeed, let $\boldsymbol{\mathcal{M}}^{fin}$ be the support region where in the search for the optimum mask will be conducted. Let $l_1^{fin} = \max\{i_1 : (i_1, i_2) \in \boldsymbol{\mathcal{M}}^{fin}\}$. Then, for all $i_1 \in [0, l_1], l_2 \leq l_1^{fin}$, the increased order filters corresponding to a single increment along a row of $\boldsymbol{\mathcal{M}}$,i.e., $\mathcal{C}_{\mathcal{M}+L(i_1)}(N_1, N_2)$ for all possible neighboring directions $\{(i_1, l_2(i_1) + 1)\}$ and or all $i_1 \in [0, l_1]$, can be estimated by applying the recursions of Table 1

$$\mathbf{C}_{\mathcal{M}}(N_1, N_2) \rightarrow \mathbf{C}_{\mathcal{M}+L(i_1)}(N_1, N_2)$$

The computational complexity of the algorithm summarized in Table 1 is $O\big((l_1 + 1)M + N_1 M\big)$ operations per recursion, where $M = \dim(\mathcal{C}_{\mathcal{M}}) = \sum_{i_1=0}^{l_1}(l_2(i_1)+1)$. Then, for a 2-D filter of a final mask shape $\boldsymbol{\mathcal{M}}^{fin}$,

$$O\Big((l_1^{fin} + 1 + N_1)(M^{fin})^2\Big)$$

operations are required.

A great advantage the proposed algorithm offers against [8], is the accommodation of masks of general boundaries and the estimation of lower order parameters. Moreover, all lower order filters that correspond to reduced shape masks can be recovered. Consider for example a filter mask of a rectangular shape, $M = (0, 0)) \times (0, \ell^{fin})$. When all filters of intermediate order $(0, 0) \times (\ell, \ell)$ are required, for all $1 \leq \ell \leq \ell^{fin}$.

3 CONCLUSIONS

A highly efficient, order recursive algorithm for 2-D FIR filtering and 2-D system identification has been developed. Masks with arbitrary shape can be handled. The proposed algorithm allows for the recursive estimation of the 2-D filter mask shape. The implicit flexibility of the algorithm enables for a dynamical reconfiguration of the mask shape in a computational efficient way. The application of the proposed scheme to 2-D image restoration and to 2-D spectral estimation are topics of current research.

References

[1] A. K. Jain, Fundamentals of Digital Image Processing, Prentice Hall, 1989.

[2] H. Kaufman and A. M. Tekalp, 'Survey of estimation techniques in image restoration,' IEEE Control Systems, pp. 16-23, Jan. 1991.

[3] B.F. McGuffin and B. Liu,'An Efficient algorithm for Two-Dimensional Autoregressive Spectrum estimation,' IEEE Trans. on ASSP-37, pp. 106-117, 1989.

[4] G. Glentis and N. Kalouptsidis, 'Fast algorithms for 2-D least squares FIR filtering,' IEEE ICASSP, 1991.

[5] G. Glentis and N. Kalouptsidis, 'Efficient multichannel FIR filtering using a step versatile order recursive algorithm,' Signal Processing, vol 37, pp. 437-462, 1994

[6] G. Glentis, C. Slump and O. Herrmann, An Efficient Algorithm for Two-Dimensional FIR filtering and System Identification, SPIE VCIP-94, Chicago, pp. 220-232, 1994.

[7] G. Glentis, C. Slump and O. Herrmann, A versatile algorithm for Two-Dimensional symmetric noncausal modeling, IEEE Trans. on Circuits and Systems, Part-II, Feb. 1998.

[8] S.L. Marple, A fast algorithm for the two-dimensional covariance method of linear prediction, Proc. ICASSP-1995, pp. 1693-1697, Detroit, USA.

[9] R. Kashyap and R. Chellappa, 'Estimation and choice of neighbors in spatial-interaction models of images,' IEEE Trans. on Information Theory, vol IT-29, pp. 60-72, Jan. 1983.

[10] P. Djuric and S. Kay, 'Model order estimation of 2D autoregressive processes,' IEEE ICASSP, pp. 3405-3408, 1991.

[11] R. Rajagopalan, M. Orchard, and K. Ramchandran, ' Optimal supports for linear predictive models', pp. 785-789, IEEE Int. Conf. Image Processing, Austin 1994.

Eq.	The Algorithm	Cost
(1)	$\beta_{L(i_1)}(N_1,N_2) = d(i_1, l_2(i_1)+1)(N_1,N_2) + \mathbf{r}_{\mathcal{M}}^{b(i_1)t}(N_1,N_2)\boldsymbol{\mathcal{C}}_{\mathcal{M}}(N_1,N_2)$	1
(2)	$\alpha^{b(i_1)}(N_1,N_2) = \rho(0,0)(N_1,N_2) + \mathbf{r}_{\mathcal{M}}^{b(i_1)t}(N_1,N_2)\mathbf{b}_{\mathcal{M}}^{i_1}(N_1,N_2)$	M
(3)	$k_{L(i_1)}(N_1,N_2) = -\beta_{L(i_1)}(N_1,N_2)/\alpha^{b(i_1)}(N_1,N_2)$	1
(4)	$\mathcal{S}_{L(i_1)}\boldsymbol{\mathcal{C}}_{\mathcal{M}+L(i_1)}(N_1,N_2) = \begin{pmatrix}\boldsymbol{\mathcal{C}}_{\mathcal{M}}(N_1,N_2)\\ 0\end{pmatrix} + \begin{pmatrix}\mathbf{b}_{\mathcal{M}}^{i_1}(N_1,N_2)\\ 1\end{pmatrix}k_{L(i_1)}(N_1,N_2)$	M
	`FOR` $\ell = 0$ `TO` l_1 `AND` $\ell \neq i_1$`, DO`	
(5)	$\beta_{L(i_1)}^{b(\ell)}(N_1,N_2) = \rho(i_1-\ell, l_2(i_1)-l_2(\ell))(N_1,N_2) + \mathbf{r}_{\mathcal{M}}^{b(i_1)t}(N_1,N_2)\mathbf{b}_{\mathcal{M}}^{\ell}(N_1,N_2)$	M
(6)	$k_{L(i_1)}^{b(\ell)}(N_1,N_2) = -\beta_{L(i_1)}^{b(\ell)}(N_1,N_2)/\alpha^{b(i_1)}(N_1,N_2)$	1
(7)	$\mathcal{S}_{L(i_1)}\mathbf{b}_{\mathcal{M}+L(i_1)}^{\ell}(N_1,N_2) = \begin{pmatrix}\mathbf{b}^\ell_{\mathcal{M}}(N_1,N_2)\\ 0\end{pmatrix} + \begin{pmatrix}\mathbf{b}_{\mathcal{M}}^{i_1}(N_1,N_2)\\ 1\end{pmatrix}k_{L(i_1)}^{b(\ell)}(N_1,N_2)$	M
	`END FOR` ℓ	
	`FOR` $n_1 = N_1$ `TO 0 DO`	
(8)	$\epsilon^{b(i_1)}(N_1, N_2-1/n_1) = x(n_1,N_2) + \boldsymbol{\mathcal{X}}_{\mathcal{M}}(n_1,N_2)^t\mathbf{b}_{\mathcal{M}}^{i_1}(N_1,N_2-1/n_1-1)$	M
(9)	$\mathbf{b}_{\mathcal{M}}^{i_1}(N_1,N_2-1/n_1) = \mathbf{b}_{\mathcal{M}}^{i_1}(N_1,N_2-1/n_1-1) - \mathbf{w}_{\mathcal{M}}(N_1,N_2-1/n_1)\epsilon^{b(i_1)}(N_1,N_2-1/n_1)$	M
(10)	$e^{b(i_1)}(N_1,N_2-1/n_1) = \epsilon^{b(i_1)}(N_1,N_2-1/n_1)\alpha\mathcal{M}+L(i_1)(N_1,N_2-1/n_1)$	1
(11)	$\alpha^{b(i_1)}(N_1,N_2-1/n_1-1) = \alpha^{b(i_1)}(N_1,N_2-1/n_1) - e^{b(i_1)}(N_1,N_2-1/n_1)\epsilon^{b(i_1)}(N_1,N_2-1/n_1)$	1
(12)	$k_{\mathcal{M}+L(i_1)}(N_1,N_2/n_1) = -e^{b(i_1)}(N_1,N_2-1/n_1)\alpha^{b(i_1)}(N_1,N_2-1/n_1-1)$	1
(13)	$\mathcal{S}_{L(i_1)}\mathbf{w}_{\mathcal{M}+L(i_1)}(N_1,N_2/n_1) = \begin{pmatrix}\mathbf{w}_{\mathcal{M}}(N_1,N_2/n_1)\\ 0\end{pmatrix}+\begin{pmatrix}\mathbf{b}^{i_1}_{\mathcal{M}}(N_1,N_2/n_1-1)\\ 1\end{pmatrix}k^{w)}_{\mathcal{M}+L(i_1)}(N_1,N_2/n_1)$	M
(14)	$\alpha\mathcal{M}+L(i_1)(N_1,N_2-1/n_1) = \alpha\mathcal{M}(N_1,N_2-1/n_1)e^{b(i_1)}(N_1,N_2-1/n_1)k^w_{\mathcal{M}+L(i_1)}(N_1,N_2/n_1)$	1
	`ENDFOR` n_1	
	`IF` $\ell = i_1$ `DO`	
(15)	`LET` $\boldsymbol{\mathcal{A}}_{\mathcal{M}} = \left[\mathbf{a}^\ell_{\mathcal{M}}(N_1,N_2)\right]_{\ell=0\ldots l_1}$	-
(16)	$\beta_L^{b(i_1)}(N_1,N_2) = \boldsymbol{\rho}_L^{b(i_1)}(N_1,N_2) + \boldsymbol{\mathcal{A}}^t_{\mathcal{M}}(N_1,N_2)\mathbf{r}^{b(i_1)}_{\mathcal{M}}(N_1,N_2)$	kM
(17)	$\boldsymbol{\alpha}^f(N_1,N_2) = R^{fo}(N_1,N_2) + \mathbf{R}^{ft}_{\mathcal{M}}(N_1,N_2)\boldsymbol{\mathcal{A}}_{\mathcal{M}}(N_1,N_2)$	kM
(18)	$K_L^{b(i_1)}(N_1,N_2) = -\boldsymbol{\alpha}^{-f}(N_1,N_2)\beta_L^{b(i_1)}(N_1,N_2)$	k^2
(19)	$\boldsymbol{\mathcal{T}}_L\mathbf{b}^{i_1}_{\mathcal{M}+L}(N_1,N_2) = \begin{pmatrix}\mathbf{0}\\ \mathbf{b}^{i_1}_{\mathcal{M}}(N_1,N_2-1)\end{pmatrix} + \begin{pmatrix}\mathbf{I}\\ \boldsymbol{\mathcal{A}}_{\mathcal{M}}(N_1,N_2)\end{pmatrix}K_L^{b(i_1)}(N_1,N_2)$	kM
(20)	`LET` $\boldsymbol{\mathcal{B}}_{\mathcal{M}+L(i_1)}(N_1,N_2) = \left[\mathbf{b}^\ell_{\mathcal{M}+L(i_1)}(N_1,N_2)\right]_{\ell=0\ldots l_1\ AND\ \ell\neq i_1}$	-
(21)	$\begin{pmatrix}\mathbf{b}^{i_1}_{\mathcal{M}+L(i_1)}(N_1,N_2)\\ \mathbf{0}\end{pmatrix} = \boldsymbol{\mathcal{S}}_R\mathbf{b}^{i_1}_{\mathcal{M}+L}(N_1,N_2) - \begin{pmatrix}\boldsymbol{\mathcal{B}}^{i_1}_{\mathcal{M}+L(i_1)}(N_1,N_2)\\ \mathbf{I}\end{pmatrix}\widehat{K}_L^{b(i_1)}(N_1,N_2)$	kM
	`ENDIF`	
	`FOR` $\ell=0$ `TO` l_1 `DO`	
(22)	$\beta^{f(\ell)}_{L(i_1)}(N_1,N_2) = \rho(i_1-\ell, l_2(i_1)+2)(N_1,N_2) + \mathbf{r}^{b(i_1)t}_{\mathcal{M}}(N_1,N_2)\mathbf{a}^\ell_{\mathcal{M}}(N_1,N_2)$	M
(23)	$k^{f(\ell)}_{L(i_1)}(N_1,N_2) = -\beta^{f(\ell)}_{L(i_1)}(N_1,N_2)/\alpha^{b(i_1)}(N_1,N_2)$	1
(24)	$\mathcal{S}^t_{L(i_1)}\mathbf{a}^\ell_{\mathcal{M}+L(i_1)}(N_1,N_2) = \begin{pmatrix}\mathbf{a}^\ell_{\mathcal{M}}(N_1,N_2-1)\\ 0\end{pmatrix}+\begin{pmatrix}\mathbf{b}^{i_1}_{\mathcal{M}}(N_1,N_2-1)\\ 1\end{pmatrix}k^{f(\ell)}_{L(i_1)}(N_1,N_2)$	M
	`END FOR` ℓ	

Table1. The true order recursive algorithm for two-dimensional Least-Squares filtering.