ELSEVIER

Signal Processing 80 (2000) 1399–1418

SIGNAL PROCESSING

www.elsevier.nl/locate/sigpro

A true order recursive algorithm for two-dimensional mean squared error linear prediction and filtering

George-Othon Glentis[a,*,1], Cornelis H. Slump[b], Otto E. Herrmann[b]

[a]TEI of Heraklion, Branch at Chania, Department of Electronics, 3, Romanou Str, Halepa, Chania 73133, Greece

[b]University of Twente, Department of Electrical Engineering (BSC), P.O. Box 217, 7500 AE Enschede, Netherlands

Received 13 February 1999; received in revised form 3 January 2000

Abstract

In this paper a novel algorithm is presented for the efficient two-dimensional (2-D), mean squared error (MSE), FIR filtering and system identification. 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 MSE linear prediction and filtering. In contrast to the existing column(row)wise 2-D recursive schemes based on the Levinson–Wiggins–Robinson multichannel algorithm, the proposed technique offers the greatest maneuverability in the 2-D index space in a computational efficient way. This flexibility can be taken into advantage if the shape of the 2-D mask is not a priori known and has to be dynamically configured. The recursive character of the algorithm allows for a continuous reshaping of the filter mask. Search for the optimal filter mask, essentially reconfigures the filter mask to achieve an optimal match. The optimum determination of the mask shape offers important advantages in 2-D system modeling, filtering and image restoration. An illustrative example from 2-D autoregressive spectrum estimation is also supplied. © 2000 Elsevier Science B.V. All rights reserved.

Zusammenfassung

In dieser Arbeit wird ein neuer Algorithmus für die effiziente zweidimensionale (2-D) FIR-Filterung und Systemidentifikation mit kleinstem mittleren Fehlerquadrat (MSE) vorgestellt. Filtermasken für allgemeine Randwerte sind zugelassen. Es werden effiziente Rekursionen für die Ordnungserhöhung entwickelt dadurch, daß die räumliche Verschiebungsinvarianz der 2-D Daten ausgenutzt wird. Die Ordnungserhöhung wird in Einzelschritten vorgenommen. Während jeder Iteration werden die Filterkoeffizienten durch ein einzelnes neues Element ergänzt. Die Verbesserungsformeln der Einzelschritte erlauben die Entwicklung eines effizienten, rein ordnungsrekursiven Algorithmus für die 2-D MSE lineare Prädiktion und Filterung. Im Gegensatz zu den existierenden spalten(reihen)-weisen 2-D rekursiven Verfahren, die auf den Levinson–Wiggins–Robinson Vielkanalalgorithmus zurückgehen, bietet die vorgeschlagenen Methode die größtmögliche Flexibilität im 2-D Indexraum für eine recheneffiziente Vorgangsweise. Diese Flexibilität kann vorteilhaft ausgenutzt werden, wenn die 2-D Maske nicht von vornherein bekannt ist und dynamisch angepaßt werden muß. Der rekursive Charakter des Algorithmus erlaubt eine stetige Anpassung der Filtermaske. Die Suche einer

* Corresponding author. Fax: + 30-821-281-90.

E-mail address: gglentis@chania.teiher.gr (G.O. Glentis).

[1] The work of Dr. Glentis was supported by the European Community, HCM and the TMR Program.

PII: S0165-1684(00)00045-1

optimalen Maske ergibt eine optimale Anpassung. Die optimale Bestimmung der Maskenform bietet wichtige Vorteile bei der 2-D Systemmodellierung, Filterung und Bildrestorierung. Ein illustratives Beispiel der Schätzung eines 2-D autoregressiven Spektrums ist beigefügt.

Résumé

Nous présentons dans cet article un algorithme nouveau pour le filtrage FIR et l'identification de systèmes bi-dimensionnels (2-D), au sens de l'erreur quadratique moyenne (MSE). Cet algorithme autorise des masques de filtres ayant des formes générales. Des récursions de mise à jour de l'ordre efficaces sont développées en exploitant la propriété d'invariance par décalage spatial de l'ensemble de données 2-D. Des récursions de mise à jour de l'ordre à un pas sont présentées. A chaque itération, l'ensembles des coefficients du filtre est augmenté d'un seul élément nouveau. Les formules de mise à jour de l'ordre à un pas permettent le développement d'un algorithme récursif efficace pour la prédiction linéaire et le filtrage MSE 2-D. Au contraire des méthodes récursives 2-D existantes opérant sur les colonnes (lignes), basées sur l'algorithme multi-canaux de Levinson–Wiggins–Robinson, la technique proposée offre la plus grande manoeuvrabilité dans l'espace des indices 2-D en étant efficace du point de vue calcul. Cette flexibilité peut être utilisée avec profit si la forme du masque 2-D n'est pas connue a priori et doit être configurée dynamiquement. Le caractère récursif de l'algorithme permet une remise en forme continue du masque du filtre. La recherche d'un masque optimal reconfigure essentiellement celui-ci pour qu'il prenne une forme optimale. La détermination optimal de la forme du masque présente des avantages certains en modélisation de systèmes 2-D, en filtrage 2-D et en restauration d'image. Un exemple illustratif sur l'estimation spectrale 2-D autorégressive est également présenté.

Keywords: 2-D FIR filtering; 2-D AR modeling; MSE estimation; Efficient algorithm

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. [11,12,14,17,18].

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*, $\mathscr{M}$:

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

Given an input 2-D signal $x(n_1, n_2)$ and a desired response 2-D signal $z(n_1, n_2)$ the MSE 2-D FIR filter is obtained by minimizing the cost function

$$\mathscr{E}[(z(n_1, n_2) - y(n_1, n_2))^2], \tag{2}$$

where $\mathscr{E}[\cdot]$ is the expectation operator. MSE 2-D linear prediction can be handled as a special case of filtering, setting $z(n_1, n_2) = x(n_1, n_2)$ and excluding the origin $\{(0, 0)\}$ from the filter mask, i.e., $(i_1, i_2) \in \mathscr{M} - \{(0, 0)\}$.

Minimization of Eq. (2) with respect to the filter parameters c_{i_1,i_2} leads to a system of linear equations, the so-called *normal equations*. Any well-behaved linear system solver can be applied for the inversion of the 2-D normal equations. 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 [18], is a well-known example. Schemes based on the Schur recursions can also be applied [15]. 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 Toeplitz-block–Toeplitz form. The column(row)wise approach, however, implies a severe restriction to system modeling since 2-D masks of rectangular shape can only be handed. Moreover, in most cases this constraint cannot be satisfied, as semicausal filtering and prediction being typical examples.

In this paper a fast algorithm is developed for the solution of the Toeplitz-block–Toeplitz normal equations in an order recursive way [3,5–7,9,10]. Filter masks of general shape are allowed. 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 MSE FIR filter and system identification, accelerating the exhaustive search procedures required by most of the order determination criteria, [1,4,13,19,20]. An example from the 2-D parametric spectrum estimation using an 2-D autoregressive model is utilized to illustrate the algorithm.

2. Support region of 2-D fir filters

In this section 2-D support regions of general shapes are considered. Thus, $\mathscr{M}$ can be strongly causal, causal, semicausal and/or noncausal. A fairly general shape for the support region is considered. Thus, $\mathscr{M}$ is allowed to be horizontally convex, i.e., the horizontal line segment joining any two points $(i_1, i_2),(i_1, i_3) \in \mathscr{M}$ lies in $\mathscr{M}$.

Consider the support region depicted in Fig. 1. More precisely, $\mathscr{M}$ consists of a union of intervals

$$\mathscr{M} = \bigcup_{i_1 = k_1}^{l_1} \boldsymbol{m}(i_1), \quad \boldsymbol{m}(i_1) = \{(i_1, i_2) : k_2(i_1) \leqslant i_2 \leqslant l_2(i_1)\}, \tag{3}$$

where $l_1 = \max\{i_1 : (i_1, i_2) \in \mathscr{M}\}$, $k_1 = \min\{i_2 : (i_1, i_2) \in \mathscr{M}\}$, $l_2(i_1) = \max\{i_2 : (i_1, i_2) \in \boldsymbol{m}(i_1)\}$, and $k_2(i_1) = \min\{i_2 : (i_1, i_2) \in \boldsymbol{m}(i_1)\}$.

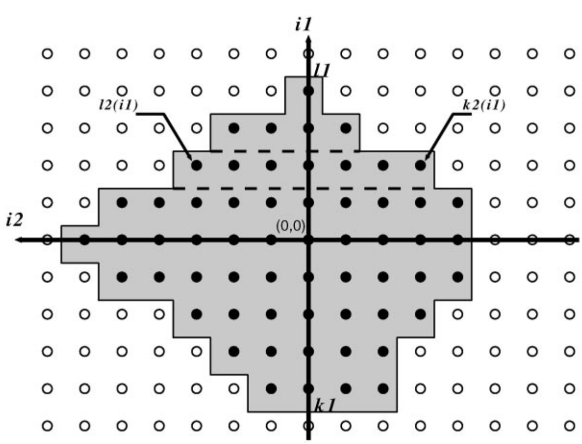

Fig. 1. Support region for a 2-D FIR filter.

Then, Eq. (1) takes the form

$$y(n_1,n_2) = -\sum_{i_1=k_1}^{l_1}\sum_{i_2=k_2(i_1)}^{l_2(i_1)} c_{i_1,i_2}x(n_1-i_1,n_2-i_2). \tag{4}$$

Let us define the data vector $\boldsymbol{x}_{m(i_1)}(n_1,n_2)$, for all $i_1\in[k_1,l_1]$, which consists of all data lying on the i_1th row, i.e., $\boldsymbol{m}(i_1)$ of the filter mask $\mathscr{M}$:

$$\boldsymbol{x}_{m(i_1)}(n_1,n_2) = [x(n_1-i_1,n_2-k_2(i_1))x(n_1-i_1,n_2-k_2(i_1)-1),$$
$$x(n_1-i_1,n_2-k_2(i_1)-2)\dots x(n_1-i_1,n_2-l_2(i_1)+1)x(n_1-i_1,n_2-l_2(i_1))]^{\mathrm{t}}.$$

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

$$\boldsymbol{c}_{m(i_1)} = [c_{i_1,k_2(i_1)}\ c_{i_1,k_2(i_1)+1}\ c_{i_1,k_2(i_1)+2}\dots c_{i_1,l_2(i_1)-1}\ c_{i_1,l_2(i_1)}]^{\mathrm{t}}.$$

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

$$\mathscr{X}_{\mathscr{M}}(n_1,n_2) = [\boldsymbol{x}^{\mathrm{t}}_{m(k_1)}(n_1,n_2)\ \boldsymbol{x}^{\mathrm{t}}_{m(k_1+1)}(n_1,n_2)\dots\boldsymbol{x}^{\mathrm{t}}_{m(l_1-1)}(n_1,n_2)\ \boldsymbol{x}^{\mathrm{t}}_{m(l_1)}(n_1,n_2)]^{\mathrm{t}},$$

$$\mathscr{C}_{\mathscr{M}} = [\boldsymbol{c}^{\mathrm{t}}_{m(k_1)}\ \boldsymbol{c}^{\mathrm{t}}_{m(k_1+1)}\ \boldsymbol{c}^{\mathrm{t}}_{m(k_1+2)}\dots\boldsymbol{c}^{\mathrm{t}}_{m(l_1-1)}\ \boldsymbol{c}^{\mathrm{t}}_{m(l_1)}]^{\mathrm{t}}. \tag{5}$$

Cleary, $\mathscr{X}_{\mathscr{M}}(n_1,n_2)$ and $\mathscr{C}_{\mathscr{M}}$ are block vectors with entry subvectors of dimensions $p(i_1)\times 1$, $i_1=k_1,\dots,l_1$, where

$$p(i_1) = l_2(i_1)-k_2(i_1)+1.$$

Thus, they both have dimensions $P\times 1$, where

$$P = \sum_{i_1=k_1}^{l_1} p(i_1). \tag{6}$$

Using definitions (5), Eq. (1) can be written in a familiar, compact, linear regression form [18]

$$y(n_1,n_2) = -\mathscr{X}^{\mathrm{t}}_{\mathscr{M}}(n_1,n_2)\mathscr{C}_{\mathscr{M}}. \tag{7}$$

The normal equations resulting from the minimization of the cost function (2) are expressed as [12,18]

$$\mathscr{R}_{\mathscr{M}}\mathscr{C}_{\mathscr{M}} = -\mathscr{D}_{\mathscr{M}}, \tag{8}$$

where $\mathscr{R}_{\mathscr{M}} = \mathscr{E}[\mathscr{X}_{\mathscr{M}}(n_1,n_2)\mathscr{X}^{\mathrm{t}}_{\mathscr{M}}(n_1,n_2)]$ and $\mathscr{D}_{\mathscr{M}} = \mathscr{E}[\mathscr{X}_{\mathscr{M}}(n_1,n_2)z(n_1,n_2)]$ are the input signal autocorrelation matrix and the cross-correlation vector between the input and the desired response signal, respectively.

In the sequel, real and homogeneous, random, wide-sense stationary 2-D signals will be considered. This implies that the autocorrelation between two samples depends on the difference of their coordinates, i.e.,

$$\mathscr{E}[x(n_1-i_1,n_2-i_2)x(n_1-j_1,n_2-j_2)] = \rho(i_1-j_1,i_2-j_2),\quad \rho(i,j)=\rho(-i,-j).$$

The autocorrelation matrix $\mathscr{R}_{\mathscr{M}}$ is a block matrix of block order l_1-k_1+1,

$$\mathscr{R}_{\mathscr{M}} = [\boldsymbol{R}(i_1,j_1)]_{\substack{i_1=k_1\dots l_1,\\ j_1=k_1\dots l_1}}\quad \boldsymbol{R}(i_1,j_1) = \mathscr{E}[\boldsymbol{x}_{m(i_1)}(n_1,n_2)\boldsymbol{x}^{\mathrm{t}}_{m(j_1)}(n_1,n_2)]. \tag{9}$$

Entries $\boldsymbol{R}(i_1, j_1)$ are Toeplitz matrices of dimensions $p(i_1) \times p(j_1)$ of the form

$$\boldsymbol{R}(i_1, j_1) = [\mathscr{E}[x(n_1 - i_1, n_2 - i_2)x(n_1 - j_1, n_2 - j_2)]]_{\substack{i_2 = k_2(i_1) \ldots l_2(i_1) \\ j_2 = k_2(j_1) \ldots l_2(j_1)}}$$

$$= [\rho(i_1 - j_1, i_2 - j_2)]_{\substack{i_2 = k_2(i_1) \ldots l_2(i_1), \\ j_2 = k_2(j_1) \ldots l_2(j_1)}} \tag{10}$$

where $\mathscr{D}_{\mathscr{M}}$ is a block vector of dimension $P \times 1$. It consists of $l_1 - k_1 + 1$ subvectors, of dimensions $p(i_1) \times 1$ each

$$\mathscr{D}_{\mathscr{M}} = [\boldsymbol{d}(i_1)]_{i_1 = k_1 \ldots l_2}, \quad \boldsymbol{d}(i_1) = [d(i_1, i_2)]_{i_2 = k_2(i_1) \ldots l_2(i_1)}, \tag{11}$$

where

$$d(i_1, i_2) = \mathscr{E}[x(n_1 - i_1, n_2 - i_2)z(n_1, n_2)]. \tag{12}$$

The two-dimensional linear prediction can be formulated in a similar way. Indeed, let us consider the one-step ahead linear predictor defined as

$$\hat{x}(n_1, n_2) = -\sum_{(i_1, i_2) \in \mathscr{M} - \{0,\ 0\}} a_{i_1, i_2} x(n_1 - i_1, n_2 - i_2). \tag{13}$$

The optimum MSE predictor coefficients a_{i_1, i_2} are obtained by minimizing the cost function

$$\mathscr{E}[(x(n_1, n_2) - \hat{x}(n_1, n_2))^2]. \tag{14}$$

It is straightforward to prove that minimization of the above cost results in the normal equations [18]

$$\mathscr{R}_{\mathscr{M}} \begin{bmatrix} 1 \\ \mathscr{A}_{\mathscr{M} - \{0,\ 0\}} \end{bmatrix} = \begin{bmatrix} E_{\min} \\ \boldsymbol{0} \end{bmatrix}, \tag{15}$$

where $E_{\min}$ is the minimum MSE attained and $\mathscr{A}_{\mathscr{M} - \{0,\ 0\}}$ is a vector that carries the predictor coefficients a_{i_1, i_2}. Linear system (15) can further be simplified by *power normalization* as

$$\mathscr{R}_{\mathscr{M}} \tilde{\mathscr{A}}_{\mathscr{M}} = \begin{bmatrix} 1 \\ \boldsymbol{0} \end{bmatrix}, \tag{16}$$

where

$$\tilde{\mathscr{A}}_{\mathscr{M}} = \begin{bmatrix} 1 \\ \mathscr{A}_{\mathscr{M} - \{0,\ 0\}} \end{bmatrix} \Big/ E_{\min}.$$

3. The proposed algorithm

In this section order updating recursions are developed for the transition from lower-order parameters to increased-order counterparts. Single-step increments of the filter mask $\mathscr{M}$ are allowed each time. Thus, starting from $\mathscr{M}$ an increased-order mask is constructed with one additional neighboring sample. Similar techniques have been proposed in the past, for the order recursive solution of the 2-D [8–10,16], and 3-D [1] Yule–Walker equations.

Let us consider the i_1th row of the filter mask, i.e., $\boldsymbol{m}(i_1)$, $i_1 \in [k_1, l_1]$. It corresponds to $p(i_1) = l_2(i_1) - k_2(i_1) + 1$ filter taps. Each row of the filter mask can be expanded by adding a new sample

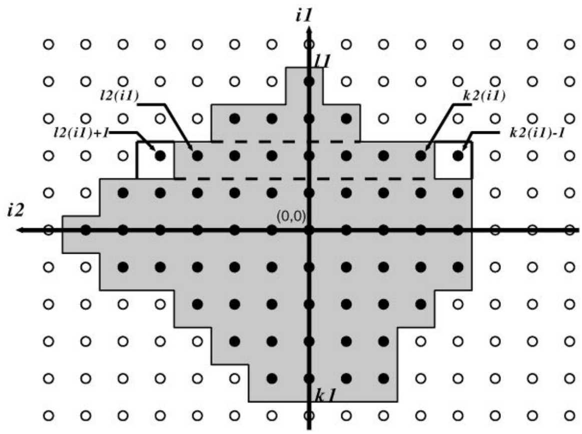

Fig. 2. Single-step size increment of the support region.

either in front or at the end of $\boldsymbol{m}(i_1)$. L and R refer to left-hand and right-hand side increment with respect to the origin $\{(0,\ 0)\}$. This is illustrated in Fig. 2. Let $\mathscr{M}+L(i_1)$ be the left-hand side increased-order filter mask. Thus,

$$\mathscr{M}+L(i_1)=\mathscr{M}\cup\{(i_1,l_2(i_1)+1)\}. \tag{17}$$

The corresponding augmented data vector (5) is partitioned as

$$\mathscr{X}_{\mathscr{M}+L(i_1)}(n_1,n_2)=\mathscr{S}^{\mathrm{t}}_{L(i_1)}\begin{bmatrix}\mathscr{X}_{\mathscr{M}}(n_1,n_2)\\ x(n_1-i_1,n_2-l_2(i_1)-1)\end{bmatrix}. \tag{18}$$

$\mathscr{S}_{L(i_1)}$ 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 $\mathscr{X}_{\mathscr{M}+L(i_1)}(n_1,n_2)$.

Similarly, let $\mathscr{M}+R(i_1)$ be the right-hand side increased-order filter mask,

$$\mathscr{M}+R(i_1)=\mathscr{M}\cup\{(i_1,k_2(i_1)-1)\}. \tag{19}$$

Then

$$\mathscr{X}_{\mathscr{M}+R(i_1)}(n_1,n_2)=\mathscr{T}^{\mathrm{t}}_{R(i_1)}\begin{bmatrix}(x(n_1-i_1,n_2-k_2(i_1)+1)\\ \mathscr{X}_{\mathscr{M}}(n_1,n_2)\end{bmatrix}, \tag{20}$$

where $\mathscr{T}_{R(i_1)}$ is a permutation matrix utilized to extract (pop-up) the extra data sample $x(n_1-i_1,n_2-k_2(i_1)+1)$ out of $\mathscr{X}_{\mathscr{M}+R(i_1)}(n_1,n_2)$.

3.1. Filter order updating recursions

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

$$\mathscr{C}_{\mathscr{M}}\to\mathscr{C}_{\mathscr{M}+L(i_1)},\quad \mathscr{C}_{\mathscr{M}}\to\mathscr{C}_{\mathscr{M}+R(i_1)}.$$

Consider the increased-order linear system (8) corresponding to the left-hand side augmented mask $\mathscr{M} + L(i_1)$. It can be partitioned using (18) as

$$\mathscr{S}^{\mathrm{t}}_{L(i_1)}\begin{bmatrix} \mathscr{R}_{\mathscr{M}} & \boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}} \\ \boldsymbol{r}^{\mathrm{b}(i_1)\mathrm{t}}_{\mathscr{M}} & \rho(0,0) \end{bmatrix}\mathscr{S}_{L(i_1)}\mathscr{C}_{\mathscr{M}+L(i_1)} = -\mathscr{S}^{\mathrm{t}}_{L(i_1)}\begin{bmatrix} \mathscr{D}_{\mathscr{M}} \\ d(i_1, l_2(i_1)+1) \end{bmatrix}, \tag{21}$$

where $d(i_1, l_2(i_1)+1) = \mathscr{E}[x(n_1 - i_1, n_2 - l_2(i_1) - 1)z(n_1, n_2)]$, and $\boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}} = \mathscr{E}[\mathscr{X}_{\mathscr{M}}(n_1, n_2)x(n_1 - i_1, n_2 - l_2(i_1) - 1)]$. Application of the matrix inversion lemma for partitioned matrices, [18], leads to a recursive way for the left-hand side order update of $\mathscr{C}_{\mathscr{M}}$,

$$\mathscr{S}_{L(i_1)}\mathscr{C}_{\mathscr{M}+L(i_1)} = \begin{pmatrix} \mathscr{C}_{\mathscr{M}} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}^{i_1}_{\mathscr{M}} \\ 1 \end{pmatrix} k_{L(i_1)}, \tag{22}$$

$$k_{L(i_1)} = -\beta_{L(i_1)}/\alpha^{\mathrm{b}(i_1)}, \tag{23}$$

$$\beta_{L(i_1)} = d(i_1, l_2(i_1)+1) + \boldsymbol{r}^{\mathrm{b}(i_1)\mathrm{t}}_{\mathscr{M}}\mathscr{C}_{\mathscr{M}}, \quad \alpha^{\mathrm{b}(i_1)} = \rho(0,0) + \boldsymbol{r}^{\mathrm{b}(i_1)\mathrm{t}}_{\mathscr{M}}\boldsymbol{b}^{i_1}_{\mathscr{M}}, \tag{24}$$

where $\boldsymbol{b}^{i_1}_{\mathscr{M}}$ is defined by

$$\mathscr{R}_{\mathscr{M}}\boldsymbol{b}^{i_1}_{\mathscr{M}} = -\boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}}. \tag{25}$$

The resulting formulae are tabulated in Table 1.

Similarly, increased order linear system corresponding to the right-hand side augmented mask $\mathscr{M} + R(i_1)$ can be partitioned as

$$\mathscr{T}^{\mathrm{t}}_{R(i_1)}\begin{bmatrix} \rho(0,0) & \boldsymbol{r}^{\mathrm{f}(i_1)\mathrm{t}}_{\mathscr{M}} \\ \boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}} & \mathscr{R}_{\mathscr{M}} \end{bmatrix}\mathscr{T}_{R(i_1)}\mathscr{C}_{\mathscr{M}+R(i_1)} = -\mathscr{T}^{\mathrm{t}}_{R(i_1)}\begin{bmatrix} d(i_1, k_2(i_1)-1) \\ \mathscr{D}_{\mathscr{M}} \end{bmatrix}, \tag{26}$$

where $\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}} = \mathscr{E}[\mathscr{X}_{\mathscr{M}}(n_1, n_2)x(n_1 - i_1, n_2 - k_2(i_1) + 1)]$.

Application of the matrix inversion lemma results in

$$\mathscr{T}_{R(i_1)}\mathscr{C}_{\mathscr{M}+R(i_1)} = \begin{pmatrix} 0 \\ \mathscr{C}_{\mathscr{M}} \end{pmatrix} + \begin{pmatrix} 1 \\ \boldsymbol{a}^{i_1}_{\mathscr{M}} \end{pmatrix} k_{R(i_1)}, \tag{27}$$

$$k_{R(i_1)} = -\beta_{R(i_1)}/\alpha^{\mathrm{f}(i_1)}, \tag{28}$$

$$\beta_{R(i_1)} = d(i_1, k_2(i_1)-1) + \boldsymbol{r}^{\mathrm{f}(i_1)\mathrm{t}}_{\mathscr{M}}\mathscr{C}_{\mathscr{M}}, \quad \alpha^{\mathrm{f}(i_1)} = r(0,0) + \boldsymbol{r}^{\mathrm{f}(i_1)\mathrm{t}}_{\mathscr{M}}\boldsymbol{a}^{i_1}_{\mathscr{M}}, \tag{29}$$

where $\boldsymbol{a}^{i_1}_{\mathscr{M}}$ is defined by

$$\mathscr{R}_{\mathscr{M}}\boldsymbol{a}^{i_1}_{\mathscr{M}} = -\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}}. \tag{30}$$

The resulting formulae are tabulated in Table 2.

The development of an order recursive algorithm for the determination of the optimum filters $\mathscr{C}_{\mathscr{M}+L(i_1)}$ or $\mathscr{C}_{\mathscr{M}+R(i_1)}$, for all possible neighboring directions $\{(i_1, l_2(i_1)+1)\}$ or $\{(i_1, k_1(i_1)-1)\}$, respectively, for all

Table 1
Left-hand side recursions

Eq.	Left-hand side recursions	Cost
(1)	$\beta_{L(i_1)} = d(i_1, l_2(i_1)+1) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{b}(i_1)\mathrm{t}}\mathscr{C}_{\mathscr{M}}$	1
(2)	$\alpha^{\mathrm{b}(i_1)} = \rho(0, 0) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{b}(i_1)\mathrm{t}}\boldsymbol{b}_{\mathscr{M}}^{i_1}$	P
(3)	$k_{L(i_1)} = -\beta_{L(i_1)}/\alpha^{\mathrm{b}(i_1)}$	1
(4)	$\mathscr{S}_{L(i_1)}\mathscr{C}_{\mathscr{M}+L(i_1)} = \left(\frac{\mathscr{C}_{\mathscr{M}}}{0}\right) + \left(\frac{\boldsymbol{b}_{\mathscr{M}}^{i_1}}{1}\right)k_{L(i_1)}$	P
	FOR $\ell = k_1$ TO l_1 AND $\ell \neq i_1$, DO	
(5)	$\beta_{L(i_1)}^{\mathrm{b}(\ell)} = \rho(i_1-\ell, l_2(i_1)-l_2(\ell)) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{b}(i_1)\mathrm{t}}\boldsymbol{b}_{\mathscr{M}}^{\ell}$	P
(6)	$k_{L(i_1)}^{\mathrm{b}(\ell)} = -\beta_{L(i_1)}^{\mathrm{b}(\ell)}/\alpha^{\mathrm{b}(i_1)}$	1
(7)	$\mathscr{S}_{L(i_1)}\boldsymbol{b}_{\mathscr{M}+L(i_1)}^{\ell} = \left(\frac{\boldsymbol{b}_{\mathscr{M}}^{\ell}}{0}\right) + \left(\frac{\boldsymbol{b}_{\mathscr{M}}^{i_1}}{1}\right)k_{L(i_1)}^{\mathrm{b}(\ell)}$	P
	END FOR ℓ	
	IF $\ell = i_1$ DO	
(8)	LET $\mathscr{A}_{\mathscr{M}} = [\boldsymbol{a}_{\mathscr{M}}^{\ell}]_{\ell=k_1,\dots,l_1}$	–
(9)	$\beta_L^{\mathrm{b}(i_1)} = \boldsymbol{\rho}_L^{\mathrm{b}(i_1)} + \mathscr{A}_{\mathscr{M}}^{\mathrm{t}}\boldsymbol{r}_{\mathscr{M}}^{\mathrm{b}(i_1)}$	k P
(10)	$\boldsymbol{\alpha}^{\mathrm{f}} = R^{\mathrm{fo}} + \boldsymbol{R}_{\mathscr{M}}^{\mathrm{ft}}\mathscr{A}_{\mathscr{M}}$	k P
(11)	$K_L^{\mathrm{b}(i_1)} = -\boldsymbol{\alpha}^{-\mathrm{f}}\beta_L^{\mathrm{b}(i_1)}$	k^2
(12)	$\mathscr{T}_L\boldsymbol{b}_{\mathscr{M}+L}^{i_1} = \left(\frac{\boldsymbol{0}}{\mathbf{b}_{\mathscr{M}}^{i_1}}\right) + \left(\frac{\boldsymbol{I}}{\mathscr{A}_{\mathscr{M}}}\right)K_L^{\mathrm{b}(i_1)}$	k P
(13)	LET $\mathscr{B}_{\mathscr{M}+L(i_1)} = [\boldsymbol{b}_{\mathscr{M}+L(i_1)}^{\ell}]_{\ell=k_1,\dots,l_1 \text{ and } \ell\neq i_1}$	–
(14)	$\left(\frac{\boldsymbol{b}_{\mathscr{M}+L(i_1)}^{i_1}}{\boldsymbol{0}}\right) = \mathscr{S}_R\boldsymbol{b}_{\mathscr{M}+L}^{i_1} - \left(\frac{\mathscr{B}_{\mathscr{M}+L(i_1)}^{i_1}}{\boldsymbol{I}}\right)\hat{K}_L^{\mathrm{b}(i_1)}$	k P
	ENDIF	
	FOR $\ell = k_1$ TO l_1 DO	
(15)	$\beta_{L(i_1)}^{\mathrm{f}(\ell)} = \rho(i_1-\ell, l_2(i_1)-k_2(\ell)+2) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{b}(i_1)\mathrm{t}}\boldsymbol{a}_{\mathscr{M}}^{\ell}$	P
(16)	$k_{L(i_1)}^{\mathrm{f}(\ell)} = -\beta_{L(i_1)}^{\mathrm{f}(\ell)}/\alpha^{\mathrm{b}(i_1)}$	1
(17)	$\mathscr{S}_{L(i_1)}^{\mathrm{t}}\boldsymbol{a}_{\mathscr{M}+L(i_1)}^{\ell} = \left(\frac{\boldsymbol{a}_{\mathscr{M}}^{\ell}}{0}\right) + \left(\frac{\boldsymbol{b}_{\mathscr{M}}^{i_1}}{1}\right)k_{L(i_1)}^{\mathrm{f}(\ell)}$	P
	END FOR ℓ	

Table 2
Right-hand side recursions

Eq.	Right-hand side recursions	Cost
(1)	$\beta_{R(i_1)} = d(i_1, k_2(i_1)-1) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{f}(i_1)\mathrm{t}}\mathscr{C}_{\mathscr{M}}$	1
(2)	$\alpha^{\mathrm{f}(i_1)} = r(0, 0) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{f}(i_1)\mathrm{t}}\boldsymbol{a}_{\mathscr{M}}^{i_1}$	P
(3)	$k_{R(i_1)} = -\beta_{R(i_1)}/\alpha^{\mathrm{f}(i_1)}$	1
(4)	$\mathscr{T}_{R(i_1)}\mathscr{C}_{\mathscr{M}+R(i_1)} = \left(\frac{0}{\mathscr{C}_{\mathscr{M}}}\right) + \left(\frac{1}{\boldsymbol{a}_{\mathscr{M}}^{i_1}}\right)k_{R(i_1)}$	P
	FOR $\ell = k_1$ TO l_1 AND $\ell \neq i_1$, DO	
(5)	$\beta_{R(i_1)}^{\mathrm{f}(\ell)} = \rho(i_1-\ell, k_2(i_1)-k_2(\ell)) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{f}(i_1)\mathrm{t}}\boldsymbol{a}_{\mathscr{M}}^{\ell}$	P
(6)	$k_{R(i_1)}^{\mathrm{f}(\ell)} = -\beta_{R(i_1)}^{\mathrm{f}(\ell)}/\alpha^{\mathrm{f}(i_1)}$	k
(7)	$\mathscr{T}_{R(i_1)}\boldsymbol{a}_{\mathscr{M}+R(i_1)}^{\ell} = \left(\frac{0}{\boldsymbol{a}_{\mathscr{M}}^{\ell}}\right) + \left(\frac{1}{\boldsymbol{a}_{\mathscr{M}}^{i_1}}\right)k_{R(i_1)}^{\mathrm{f}(\ell)}$	P
	END FOR ℓ	
	IF $\ell = i_1$ DO	
(8)	LET $\mathscr{B}_{\mathscr{M}} = [\boldsymbol{b}_{\mathscr{M}}^{\ell}]_{\ell=k_1,\dots,l_1}$	–
(9)	$\beta_R^{\mathrm{f}(i_1)} = \boldsymbol{\rho}_R^{\mathrm{f}(i_1)} + \mathscr{B}_{\mathscr{M}}^{\mathrm{t}}\boldsymbol{r}_{\mathscr{M}}^{\mathrm{f}(i_1)}$	k
(10)	$\boldsymbol{\alpha}^{\mathrm{b}} = R^{\mathrm{b}o} + \boldsymbol{R}_{\mathscr{M}}^{\mathrm{bt}}\mathscr{B}_{\mathscr{M}}$	k P
(11)	$K_R^{\mathrm{f}(i_1)} = -\boldsymbol{\alpha}^{-\mathrm{b}}\beta_R^{\mathrm{f}(i_1)}$	k^2
(12)	$\mathscr{S}_R\boldsymbol{a}_{\mathscr{M}+R}^{i_1} = \left(\frac{\boldsymbol{a}_{\mathscr{M}}^{i_1}}{\boldsymbol{0}}\right) + \left(\frac{\mathscr{B}_{\mathscr{M}}}{\boldsymbol{I}}\right)K_R^{\mathrm{f}(i_1)}$	k P
(13)	LET $\mathscr{A}_{\mathscr{M}+R(i_1)} = [\boldsymbol{a}_{\mathscr{M}+R(i_1)}^{\ell}]_{\ell=k_1,\dots,l_1 \text{ and } \ell\neq i_1}$	–
(14)	$\left(\frac{\boldsymbol{0}}{\boldsymbol{a}_{\mathscr{M}+R(i_1)}^{i_1}}\right) = \mathscr{T}_R\boldsymbol{a}_{\mathscr{M}+R}^{i_1} - \left(\frac{\boldsymbol{I}}{\mathscr{A}_{\mathscr{M}+R(i_1)}^{i_1}}\right)\hat{K}_R^{\mathrm{f}(i_1)}$	k P
	ENDIF	
	FOR $\ell = k_1$ TO l_1 DO	
(15)	$\beta_{R(i_1)}^{\mathrm{b}(\ell)} = \rho(i_1-\ell, k_2(i_1)-l_2(\ell)-2) + \boldsymbol{r}_{\mathscr{M}}^{\mathrm{f}(i_1)\mathrm{t}}\boldsymbol{b}_{\mathscr{M}}^{\ell}$	k P
(16)	$k_{R(i_1)}^{\mathrm{b}(\ell)} = -\beta_{R(i_1)}^{\mathrm{b}(\ell)}/\alpha^{\mathrm{f}(i_1)}$	P
(17)	$\mathscr{T}_{R(i_1)}\boldsymbol{b}_{\mathscr{M}+R(i_1)}^{\ell} = \left(\frac{0}{\boldsymbol{b}_{\mathscr{M}}^{\ell}}\right) + \left(\frac{1}{\boldsymbol{a}_{\mathscr{M}}^{i_1}}\right)k_{R(i_1)}^{\mathrm{b}(\ell)}$	k P
	END FOR ℓ	

$i_1 \in [k_1, l_1]$, requires recursions for updating $\boldsymbol{b}^{\ell}_{\mathscr{M}}$ and $\boldsymbol{a}^{\ell}_{\mathscr{M}}$ for all $\ell = k_1, \dots, l_1$. These are backward and forward predictors that minimize the cost functions

$$\mathscr{E}[(x(n_1-\ell, n_2-l_2(\ell)-1)+\mathscr{X}^{\mathrm{t}}_{\mathscr{M}}(n_1,n_2)\boldsymbol{b}^{\ell}_{\mathscr{M}})^2],$$

$$\mathscr{E}[(x(n_1-\ell, n_2-k_2(\ell)+1)+\mathscr{X}^{\mathrm{t}}_{\mathscr{M}}(n_1,n_2)\boldsymbol{a}^{\ell}_{\mathscr{M}})^2].$$

They are estimated as the solutions of the normal equations

$$\mathscr{R}_{\mathscr{M}}\boldsymbol{b}^{\ell}_{\mathscr{M}} = -\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}}, \quad \mathscr{R}_{\mathscr{M}}\boldsymbol{a}^{\ell}_{\mathscr{M}} = -\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}}, \quad \ell \in [k_1, l_1], \tag{31}$$

where the backward and forward autocorrelation vectors are defined as

$$\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}} = \mathscr{E}[\mathscr{X}_{\mathscr{M}}(n_1,n_2)x(n_1-\ell, n_2-l_2(\ell)-1)],$$

$$\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}} = \mathscr{E}[\mathscr{X}_{\mathscr{M}}(n_1,n_2)x(n_1-\ell, n_2-k_2(\ell)+1)].$$

To be able to develop an order recursive algorithm for the determination of the optimum filter $\mathscr{C}_{\mathscr{M}}$, left-hand, as well as, right-hand side recursions for updating the single-channel backward and forward predictors are required. To this end, left-hand and right-hand side, multiple-step increment support regions, necessary for the efficient order updating of the forward and backward predictors, are first defined.

3.2. Multiple-step increment of the support region

The increased order masks $\mathscr{M}+L(i_1)$ and $\mathscr{M}+R(i_1)$ considered so far, correspond to a single-order increment of the filter configuration, either to the left- or to the right-hand side, along the i_1th row of the filter mask. In the sequel, block step increments of the filter masks will be developed.

The left-hand, as well as the right-hand, side block increased masks are defined as follows (see Fig. 3):

$$\mathscr{M}+\boldsymbol{L} = \mathscr{M} \bigcup_{i_1=k_1}^{l_1} \{(i_1, l_2(i_1)+1)\}, \tag{32}$$

$$\mathscr{M}+\boldsymbol{R} = \mathscr{M} \bigcup_{i_1=k_1}^{l_1} \{(i_1, k_2(i_1)-1)\}. \tag{33}$$

The corresponding data vectors can be partitioned as

$$\mathscr{X}_{\mathscr{M}+\boldsymbol{L}}(n_1,n_2) = \mathscr{T}^{\mathrm{t}}_{\boldsymbol{L}}\begin{bmatrix} \boldsymbol{x}^{\mathrm{f}}(n_1,n_2-1) \\ \mathscr{X}_{\mathscr{M}}(n_1,n_2-1)\end{bmatrix} = \mathscr{S}^{\mathrm{t}}_{\boldsymbol{L}}\begin{bmatrix} \mathscr{X}_{\mathscr{M}+L(i_1)}(n_1,n_2) \\ \boldsymbol{x}^{\mathrm{b}}_{i_1}(n_1,n_2)\end{bmatrix}, \tag{34}$$

$$\mathscr{X}_{\mathscr{M}+\boldsymbol{R}}(n_1,n_2) = \mathscr{S}^{\mathrm{t}}_{\boldsymbol{R}}\begin{bmatrix} \mathscr{X}_{\mathscr{M}}(n_1,n_2+1) \\ \boldsymbol{x}^{\mathrm{b}}(n_1,n_2+1)\end{bmatrix} = \mathscr{T}^{\mathrm{t}}_{\boldsymbol{R}}\begin{bmatrix} \boldsymbol{x}^{\mathrm{f}}_{i_1}(n_1,n_2) \\ \mathscr{X}_{\mathscr{M}+R(i_1)}(n_1,n_2)\end{bmatrix}. \tag{35}$$

The vectors $\boldsymbol{x}^{\mathrm{b}}(n_1,n_2)$ and $\boldsymbol{x}^{\mathrm{f}}(n_1,n_2)$ have dimensions $(l_1-k_1+1)\times 1$ and consist of all left-(right-) hand side extra samples of the extended filter masks,

$$\begin{aligned}\boldsymbol{x}^{\mathrm{b}}(n_1,n_2) = [&x(n_1-k_1, n_2-l_2(k_1)-1)\ x(n_1-k_1+1, n_2-l_2(k_1+1)-1)\dots \\ &x(n_1-l_1+1, n_2-l_2(l_1-1)-1)x(n_1-l_1, n_2-l_2(l_1)-1)]^{\mathrm{t}},\end{aligned} \tag{36}$$

$$\begin{aligned}\boldsymbol{x}^{\mathrm{f}}(n_1,n_2) = [&x(n_1-k_1, n_2-k_2(k_1)+1)x(n_1-k_1+1, n_2-k_2(k_1+1)+1)\dots \\ &x(n_1-l_1+1, n_2-k_2(l_1-1)+1)x(n_1-l_1, n_2-k_2(l_1)+1)]^{\mathrm{t}}.\end{aligned} \tag{37}$$

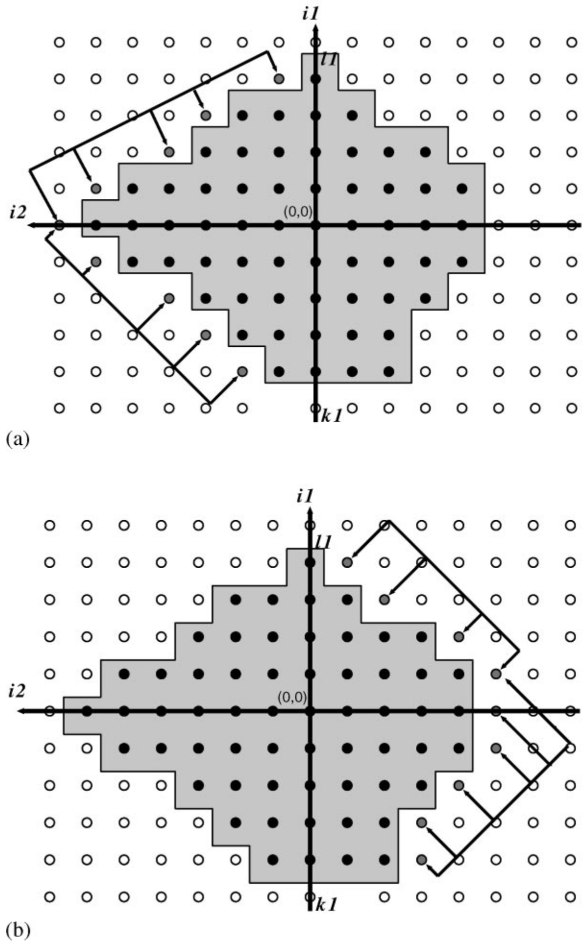

Fig. 3. Block increment of the 2-D support region: (a) left-hand side block increment, (b) right-hand side block increment.

Th vectors $\boldsymbol{x}^{\mathrm{b}}_{i_1}(n_1,n_2)$ and $\boldsymbol{x}^{\mathrm{f}}_{i_1}(n_1,n_2)$ are obtained by extracting the i_1th data sample $x(n_1-i_1,n_2-l_2(i_1)-1)$, or $x(n_1-i_1,n_2-k_2(i_1)+1)$, out of $\boldsymbol{x}^{\mathrm{b}}(n_1,n_2)$, or $\boldsymbol{x}^{\mathrm{f}}(n_1,n_2)$, respectively, i.e.,

$$\boldsymbol{x}^{\mathrm{b}}(n_1,n_2)=S^{\mathrm{t}}\begin{bmatrix}\boldsymbol{x}^{\mathrm{b}}_{i_1}(n_1,n_2)\\ x(n_1-i_1,n_2-l_2(i_1)-1)\end{bmatrix},\qquad \boldsymbol{x}^{\mathrm{f}}(n_1,n_2)=T^{\mathrm{t}}\begin{bmatrix}x(n_1-i_1,n_2-k_2(i_1)+1))\\ \boldsymbol{x}^{\mathrm{f}}_{i_1}(n_1,n_2)\end{bmatrix},$$

where $\mathscr{T}_R$,$\mathscr{S}_R$, $\mathscr{T}_L\mathscr{S}_L$, S and T are suitable permutation matrices.

3.3. Left-hand side predictors order updating recursions

Let us consider the left-hand side increased-order backward predictors corresponding to a single-step mask increment along the i_1th row

$$\mathscr{R}_{\mathscr{M}+L(i_1)}\boldsymbol{b}^{\ell}_{\mathscr{M}+L(i_1)}=-\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}+L(i_1)},\quad \ell\in[k_1,l_1], \tag{38}$$

where

$$\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}+L(i_1)} = \mathscr{E}[\mathscr{X}_{\mathscr{M}+L(i_1)}(n_1,n_2)x(n_1-\ell,n_2-l_2(\ell)-1-\delta(\ell-i_1))], \tag{39}$$

with $\delta(n)$ being the Kronecker delta function. Then, for all $\ell \neq i_1$, $\ell \in [k_1, l_1]$, $\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}+L(i_1)}$ is partitioned as

$$\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}+L(i_1)} = S^{\mathrm{t}}_{L(i_1)}\begin{bmatrix} \boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}} \\ \rho(i_1-\ell, l_2(i_1)-l_2(\ell)) \end{bmatrix}. \tag{40}$$

For $\ell \neq i_1$, the increased-order backward predictors are updated similar to $\mathscr{C}_{\mathscr{M}+L(i_1)}$. Thus,

$$\mathscr{S}_{L(i_1)}\boldsymbol{b}^{\ell}_{\mathscr{M}+L(i_1)} = \begin{pmatrix} \boldsymbol{b}^{\ell}_{\mathscr{M}} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}^{i_1}_{\mathscr{M}} \\ 1 \end{pmatrix} k^{\mathrm{b}(\ell)}_{L(i_1)}, \tag{41}$$

$$k^{\mathrm{b}(\ell)}_{L(i_1)} = -\beta^{\mathrm{b}(\ell)}_{L(i_1)}/\alpha^{\mathrm{b}(i_1)}, \quad \beta^{\mathrm{b}(\ell)}_{L(i_1)} = \rho(i_1-\ell, l_2(i_1)-l_2(\ell)) + \boldsymbol{r}^{\mathrm{b}(i_1)\mathrm{t}}_{\mathscr{M}}\boldsymbol{b}^{\ell}_{\mathscr{M}}. \tag{42}$$

The case $\ell = i_1$ is treated separately. It cannot be handled using a direct single-step recursion, because the extra delay introduced along the i_1th row results in incompatible data partitions. To overcome this difficulty, a multi-step-up following by a multi-step-down recursive estimation will be utilized. Indeed, consider the increased order system associated with the multi-step augmented mask (32),

$$\mathscr{R}_{\mathscr{M}+L}\boldsymbol{b}^{i_1}_{\mathscr{M}+L} = -\boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}+L}. \tag{43}$$

Matrix $\mathscr{R}_{\mathscr{M}+L}$ is partitioned using (34) as

$$\mathscr{R}_{\mathscr{M}+L} = \mathscr{T}^{\mathrm{t}}_{L}\begin{bmatrix} R^{\mathrm{fo}} & \boldsymbol{R}^{\mathrm{ft}}_{\mathscr{M}} \\ \boldsymbol{R}^{\mathrm{f}}_{\mathscr{M}} & \mathscr{R}_{\mathscr{M}} \end{bmatrix}\mathscr{T}_{L} = \mathscr{S}^{\mathrm{t}}_{L}\begin{bmatrix} \mathscr{R}_{\mathscr{M}+L(i_1)} & \hat{\boldsymbol{R}}^{\mathrm{b}(i_1)}_{\mathscr{M}} \\ \hat{\boldsymbol{R}}^{\mathrm{b}(i_1)\mathrm{t}}_{\mathscr{M}} & \hat{R}^{\mathrm{b}o}_{i_1} \end{bmatrix}\mathscr{S}_{L}, \tag{44}$$

where

$$\boldsymbol{R}^{\mathrm{f}}_{\mathscr{M}} = [\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}}]_{\ell=k_1,\dots,l_1}, \quad \hat{\boldsymbol{R}}^{\mathrm{b}(i_1)}_{\mathscr{M}} = [\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}}]_{\substack{\ell=k_1,\dots,l_1 \\ \text{and } \ell\neq i_1}}. \tag{45}$$

Small matrices appearing the top and on the bottom corner are defined as

$$R^{fo} = \mathscr{E}[\boldsymbol{x}^{\mathrm{f}}(n_1,n_2)\boldsymbol{x}^{\mathrm{ft}}(n_1,n_2)] = [\rho(\ell_1-\ell_2, k_2(\ell_1)-k_2(\ell_2)]_{\substack{\ell_1=k_1,\dots,l_2, \\ \ell_2=k_1,\dots,l_1}} \tag{46}$$

$$\hat{R}^{\mathrm{b}o}_{i_1} = \mathscr{E}[\boldsymbol{x}^{\mathrm{b}}_{i_1}(n_1,n_2)\boldsymbol{x}^{\mathrm{bt}}_{i_1}(n_1,n_2)] = [\rho(\ell_1-\ell_2, l_2(\ell_1)-l_2(\ell_2)]_{\substack{\ell_1=k_1,\dots,l_2 \text{ and } \ell_1\neq i_1 \\ \ell_2=k_1,\dots,l_1 \text{ and } \ell_2\neq i_1}}. \tag{47}$$

The vector $\boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}+L}$ is compatibly partitioned as

$$\boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}+L} = \mathscr{T}^{\mathrm{t}}_{L}\begin{bmatrix} \boldsymbol{\rho}^{\mathrm{b}(i_1)}_{L} \\ \boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}} \end{bmatrix} = \mathscr{S}^{\mathrm{t}}_{L}\begin{bmatrix} \boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}+L(i_1)} \\ \hat{\boldsymbol{\rho}}^{\mathrm{b}(i_1)}_{L} \end{bmatrix}, \tag{48}$$

where

$$\boldsymbol{\rho}^{\mathrm{b}(i_1)}_{L} = [\rho(\ell-i_1, k_2(\ell)-l_2(i_1)-2)]_{\ell=k_1,\dots,l_1}, \tag{49}$$

$$\hat{\boldsymbol{\rho}}^{\mathrm{b}(i_1)}_{L} = [\rho(\ell-i_1, l_2(\ell)-l_2(i_1)-1)]_{\ell=k_1,\dots,l_1 \text{ and } \ell\neq i_1}. \tag{50}$$

Application of the matrix inversion lemma yields

$$\mathscr{T}_L \boldsymbol{b}^{i_1}_{\mathscr{M}+L} = \begin{pmatrix} \mathbf{0} \\ \boldsymbol{b}^{i_1}_{\mathscr{M}} \end{pmatrix} + \begin{pmatrix} \boldsymbol{I} \\ \mathscr{A}_{\mathscr{M}} \end{pmatrix} K_L^{\mathrm{b}(i_1)} \quad \mathscr{S}_R \boldsymbol{b}^{i_1}_{\mathscr{M}+L} = \begin{pmatrix} \boldsymbol{b}^{i_1}_{\mathscr{M}+L(i_1)} \\ \mathbf{0} \end{pmatrix} + \begin{pmatrix} \mathscr{B}^{i_1}_{\mathscr{M}+L(i_1)} \\ \boldsymbol{I} \end{pmatrix} \hat{K}_L^{\mathrm{b}(i_1)}, \tag{51}$$

where

$$\mathscr{A}_{\mathscr{M}} = [\boldsymbol{a}^{k_1}_{\mathscr{M}} \ \boldsymbol{a}^{k_1+1}_{\mathscr{M}} \ldots \boldsymbol{a}^{l_1-1}_{\mathscr{M}} \ \boldsymbol{a}^{l_1}_{\mathscr{M}}],$$

$$K_L^{\mathrm{b}(i_1)} = -\boldsymbol{\alpha}^{-\mathrm{f}} \beta_L^{\mathrm{b}(i_1)}, \quad \beta_L^{\mathrm{b}(i_1)} = \boldsymbol{\rho}_L^{\mathrm{b}(i_1)} + \mathscr{A}^{\mathrm{t}}_{\mathscr{M}} \boldsymbol{r}^{\mathrm{b}(i_1)}_{\mathscr{M}}, \quad \boldsymbol{\alpha}^{\mathrm{f}} = R^{\mathrm{f}o} + \boldsymbol{R}^{\mathrm{ft}}_{\mathscr{M}} \mathscr{A}_{\mathscr{M}} \tag{52}$$

and

$$\mathscr{B}^{i_1}_{\mathscr{M}+L(i_1)} = [\boldsymbol{b}^{k_1}_{\mathscr{M}} \ldots \boldsymbol{b}^{i_1-1}_{\mathscr{M}} \ \boldsymbol{b}^{i_1+1}_{\mathscr{M}} \ldots \boldsymbol{b}^{l_1}_{\mathscr{M}}].$$

The parameter $\hat{K}_L^{\mathrm{b}(i_1)}$ is defined in a similar way as in Eq. (69). However, once $\mathscr{T}_L \boldsymbol{b}^{i_1}_{\mathscr{M}+L}$ has been estimated, $\hat{K}_L^{\mathrm{b}(i_1)}$ is directly obtained as the lowest $k \times 1$ part of $\mathscr{S}_L \boldsymbol{b}^{i_1}_{\mathscr{M}+L}$.

The left-hand side forward predictors order updating recursions can be derived taking into account the data partitions defined by Eq. (25). Indeed, let $\boldsymbol{a}^{\ell}_{\mathscr{M}+L(i_1)}$ be the increased-order predictor along the i_1th row, i.e.,

$$\mathscr{R}_{\mathscr{M}+L(i_1)} \boldsymbol{a}^{\ell} = -\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}+L(i_1)}. \tag{53}$$

The parameter $\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}+L(i_1)}$ is partitioned as

$$\boldsymbol{r}^{\mathrm{f}\ell}_{\mathscr{M}+L(i_1)} = \mathscr{S}^{\mathrm{t}}_{L(i_1)} \begin{bmatrix} \boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}} \\ \rho(i_1-\ell, l_2(i_1) - k_2(\ell) + 2) \end{bmatrix}. \tag{54}$$

Thus,

$$\mathscr{S}^{\mathrm{t}}_{L(i_1)} \boldsymbol{a}^{\ell}_{\mathscr{M}+L(i_1)} = \begin{pmatrix} \boldsymbol{a}^{\ell}_{\mathscr{M}} \\ 0 \end{pmatrix} + \begin{pmatrix} \boldsymbol{b}^{i_1}_{\mathscr{M}} \\ 1 \end{pmatrix} k^{\mathrm{f}(\ell)}_{L(i_1)}, \tag{55}$$

where

$$k^{\mathrm{f}(\ell)}_{L(i_1)} = -\beta^{\mathrm{f}(\ell)}_{L(i_1)} / \alpha^{\mathrm{b}(i_1)}, \quad \beta^{\mathrm{f}(\ell)}_{L(i_1)} = \rho(i_1-\ell, l_2(i_1) - k_2(\ell) + 2) + \boldsymbol{r}^{\mathrm{b}(i_1)\mathrm{t}}_{\mathscr{M}} \boldsymbol{a}^{\ell}_{\mathscr{M}}. \tag{56}$$

The resulting formulae are tabulated in Table 1.

3.4. Right-hand side predictors order updating recursions

Let us consider the right-hand side increased-order forward predictors corresponding to a single-step mask increment along the i_1th row

$$\mathscr{R}_{\mathscr{M}+R(i_1)} \boldsymbol{a}^{\ell}_{\mathscr{M}+L(i_1)} = -\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}+R(i_1)}, \quad \ell \in [k_1, l_1], \tag{57}$$

where

$$\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}+R(i_1)} = \mathscr{E}[\mathscr{X}_{\mathscr{M}+R(i_1)}(n_1,n_2) x(n_1 - i_1, n_2 - k_2(i_1) + 1 + \delta(\ell - i_1))]. \tag{58}$$

Then, for all $\ell \neq i_1$, $\ell \in [k_1, l_1]$, the increased-order forward predictors are updated similar to the $\mathscr{C}_{\mathscr{M}+R(i_1)}$ vector. Thus,

$$\mathscr{T}_{R(i_1)} \boldsymbol{a}^{\ell}_{\mathscr{M}+R(i_1)} = \begin{pmatrix} 0 \\ \boldsymbol{a}^{\ell}_{\mathscr{M}} \end{pmatrix} + \begin{pmatrix} 1 \\ \boldsymbol{a}^{i_1}_{\mathscr{M}} \end{pmatrix} k^{\mathrm{f}(\ell)}_{R(i_1)}, \tag{59}$$

$$k^{\mathrm{f}(\ell)}_{R(i_1)} = -\beta^{\mathrm{f}(\ell)}_{R(i_1)} / \alpha^{\mathrm{f}(i_1)}, \quad \beta^{\mathrm{f}(\ell)}_{R(i_1)} = \rho(i_1-\ell, k_2(i_1) - k_2(\ell)) + \boldsymbol{r}^{\mathrm{f}(i_1)\mathrm{t}}_{\mathscr{M}} \boldsymbol{a}^{\ell}_{\mathscr{M}}. \tag{60}$$

The case $\ell = i_1$ is handled similarly to the left-hand side backward predictors. Indeed, let

$$\mathscr{R}_{\mathscr{M}+R}\boldsymbol{a}^{i_1}_{\mathscr{M}+R} = -\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}+R}. \tag{61}$$

The matrix $\mathscr{R}_{\mathscr{M}+\mathrm{L}}$ is partitioned using (31) as

$$\mathscr{R}_{\mathscr{M}+R} = \mathscr{S}^{\mathrm{t}}_{R}\begin{bmatrix}\mathscr{R}_{\mathscr{M}} & \boldsymbol{R}^{\mathrm{b}}_{\mathscr{M}}\\ \boldsymbol{R}^{\mathrm{bt}}_{\mathscr{M}} & R^{\mathrm{b}o}\end{bmatrix}\mathscr{S}_{R} = \mathscr{T}^{\mathrm{t}}_{R}\begin{bmatrix}\hat{R}^{\mathrm{f}o}_{i_1} & \hat{\boldsymbol{R}}^{\mathrm{f}(i_1)\mathrm{t}}_{\mathscr{M}}\\ \hat{\boldsymbol{R}}^{\mathrm{f}(i_1)}_{\mathscr{M}} & \mathscr{R}_{\mathscr{M}+\mathrm{L}(i_1)}\end{bmatrix}\mathscr{T}_{R}, \tag{62}$$

where

$$\boldsymbol{R}^{\mathrm{b}}_{\mathscr{M}} = [\boldsymbol{r}^{\mathrm{b}(\ell)}_{\mathscr{M}}]_{\ell=k_1,\dots,l_1}, \quad \hat{\boldsymbol{R}}^{\mathrm{f}(i_1)}_{\mathscr{M}} = [\boldsymbol{r}^{\mathrm{f}(\ell)}_{\mathscr{M}}]_{\substack{\ell=k_1,\dots,l_1\\ \text{and }\ell\neq i_1}} \tag{63}$$

and

$$R^{\mathrm{b}o} = [\rho(\ell_1-\ell_2, l_2(\ell_1)-l_2(\ell_2))]_{\substack{\ell_1=k_1,\dots,l_2,\\ \ell_2=k_1,\dots,l_1}}$$

$$\hat{R}^{\mathrm{f}o}_{i_1} = [\rho(\ell_1-\ell_2, k_2(\ell_1)-k_2(\ell_2))]_{\substack{\ell_1=k_1,\dots,l_2 \text{ and } \ell_1\neq i_1,\\ \ell_2=k_1,\dots,l_1 \text{ and } \ell_2\neq i_1}} \tag{64}$$

$\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}+R}$ is compatibly partitioned as

$$\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}+R} = \mathscr{S}^{\mathrm{t}}_{R}\begin{bmatrix}\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}}\\ \boldsymbol{\rho}^{\mathrm{f}(i_1)}_{R}\end{bmatrix} = \mathscr{T}^{\mathrm{t}}_{R}\begin{bmatrix}\hat{\boldsymbol{\rho}}^{\mathrm{f}(i_1)}_{R}\\ \boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}+R(i_1)}\end{bmatrix}, \tag{65}$$

where

$$\boldsymbol{\rho}^{\mathrm{f}(i_1)}_{R} = [\rho(\ell-i_1, l_2(\ell)-k_2(i_1)-2)]_{\ell=k_1,\dots,l_1}, \tag{66}$$

$$\hat{\boldsymbol{\rho}}^{\mathrm{f}(i_1)}_{R} = [\rho(\ell-i_1, k_2(\ell)-k_2(i_1)-1)]_{\ell=k_1,\dots,l_1 \text{ and } \ell\neq i_1}. \tag{67}$$

Then the application of the matrix inversion lemma yields

$$\mathscr{S}_{R}\boldsymbol{a}^{i_1}_{\mathscr{M}+R} = \begin{pmatrix}\boldsymbol{a}^{i_1}_{\mathscr{M}}\\ \boldsymbol{0}\end{pmatrix} + \begin{pmatrix}\mathscr{B}_{\mathscr{M}}\\ \boldsymbol{I}\end{pmatrix}K^{\mathrm{f}(i_1)}_{R}, \quad \mathscr{T}_{R}\boldsymbol{a}^{i_1}_{\mathscr{M}+R} = \begin{pmatrix}\boldsymbol{0}\\ \boldsymbol{a}^{i_1}_{\mathscr{M}+R(i_1)}\end{pmatrix} + \begin{pmatrix}\boldsymbol{I}\\ \mathscr{A}^{i_1}_{\mathscr{M}+R(i_1)}\end{pmatrix}\hat{K}^{\mathrm{f}(i_1)}_{R}, \tag{68}$$

where

$$\mathscr{B}_{\mathscr{M}} = [\boldsymbol{b}^{k_1}_{\mathscr{M}}\boldsymbol{b}^{k_1+1}_{\mathscr{M}}\dots\boldsymbol{b}^{l_1-1}_{\mathscr{M}}\boldsymbol{b}^{l_1}_{\mathscr{M}}],$$

$$K^{\mathrm{f}(i_1)}_{R} = -\boldsymbol{\alpha}^{-b}\beta^{\mathrm{f}(i_1)}_{R}, \quad \beta^{\mathrm{f}(i_1)}_{R} = \boldsymbol{\rho}^{\mathrm{f}(i_1)}_{R} + \mathscr{B}^{\mathrm{t}}_{\mathscr{M}}\boldsymbol{r}^{\mathrm{f}(i_1)}_{\mathscr{M}}, \quad \boldsymbol{\alpha}^{\mathrm{b}} = R^{\mathrm{b}o} + \boldsymbol{R}^{\mathrm{bt}}_{\mathscr{M}}\mathscr{B}_{\mathscr{M}} \tag{69}$$

and

$$\mathscr{A}^{i_1}_{\mathscr{M}+R(i_1)} = [\boldsymbol{a}^{k_1}_{\mathscr{M}}\dots\boldsymbol{a}^{i_1-1}_{\mathscr{M}}\boldsymbol{a}^{i_1+1}_{\mathscr{M}}\dots\boldsymbol{a}^{l_1}_{\mathscr{M}}].$$

The right-hand side recursions for the backward predictors are similar to (43), i.e.,

$$\mathscr{T}_{R(i_1)}\boldsymbol{b}^{\ell}_{\mathscr{M}+R(i_1)} = \begin{pmatrix}0\\ \boldsymbol{b}^{\ell}_{\mathscr{M}}\end{pmatrix} + \begin{pmatrix}1\\ \boldsymbol{a}^{i_1}_{\mathscr{M}}\end{pmatrix}k^{\mathrm{b}(\ell)}_{R(i_1)}, \tag{70}$$

where

$$k^{\mathrm{b}(\ell)}_{R(i_1)} = -\beta^{\mathrm{b}(\ell)}_{R(i_1)}/\alpha^{\mathrm{f}(i_1)}, \quad \beta^{\mathrm{b}(\ell)}_{R(i_1)} = \rho(i_1-\ell, k_2(i_1)-l_2(\ell)-2) + \boldsymbol{r}^{\mathrm{f}(i_1)\mathrm{t}}_{\mathscr{M}}\boldsymbol{b}^{\ell}_{\mathscr{M}}. \tag{71}$$

The resulting formulae are tabulated in Table 2.

3.5. Overall organization

The order recursive equations developed so far, for the left-hand or the right-hand side updating of the filter coefficient vectors, as well as for the auxiliary backward and forward single-step predictors, can be tied together to form a powerful *true* order recursive 2-D algorithm. Indeed, let $\mathscr{M}^{\text{fin}}$ be the support region wherein the search for the optimum mask will be conducted. Let $k_1^{\text{fin}} = \min\{i_1 : (i_1, i_2) \in \mathscr{M}^{\text{fin}}\}$, $l_1^{\text{fin}} = \max\{i_1 : (i_1, i_2) \in \mathscr{M}^{\text{fin}}\}$. Then, for all $i_1 \in [k_1, l_1], k_1 \geqslant k_1^{\text{fin}}, l_2 \leqslant l_1^{\text{fin}}$, the increased-order filters corresponding to a single increment along a row of $\mathscr{M}$, i.e., $\mathscr{C}_{\mathscr{M}+L(i_1)}$ or $\mathscr{C}_{\mathscr{M}+R(i_1)}$, for all possible neighboring directions $\{(i_1, l_2(i_1)+1)\}$ or $\{(i_1, k_1(i_1)-1)\}$, respectively, and or all $i_1 \in [k_1, l_1]$, can be estimated by applying either the left- or the right-hand side recursions

$$\boldsymbol{C}_{\mathscr{M}} \to \boldsymbol{C}_{\mathscr{M}+L(i_1)}, \quad \boldsymbol{C}_{\mathscr{M}} \to \boldsymbol{C}_{\mathscr{M}+R(i_1)}. \tag{72}$$

The update of parameters to a mask that contains an extra row, i.e. going from $k_1 \to k_1 - 1$ or from $l_1 \to l_1 + 1$, can be accomplished only for the points lying across the vertical axis, i.e., $\{(k_1 - 1, 0)\}$ or $\{(l_1 + 1, 0)\}$, using either left of right-hand side recursions. Starting from estimating the increased-order filter that corresponds to $\mathscr{M} \cup \{(k_1 - 1, 0)\}$ or $\mathscr{M} \cup \{(l_1 + 1, 0)\}$ is determined, the remaining recursions along this row can be performed in either directions, using the left- or the right-hand side recursive equations, respectively.

The order updating procedure is illustrated using two simple but important support regions depicted in Fig. 4. Let us first consider the strongly causal square mask. Suppose that the MSE filter corresponding to

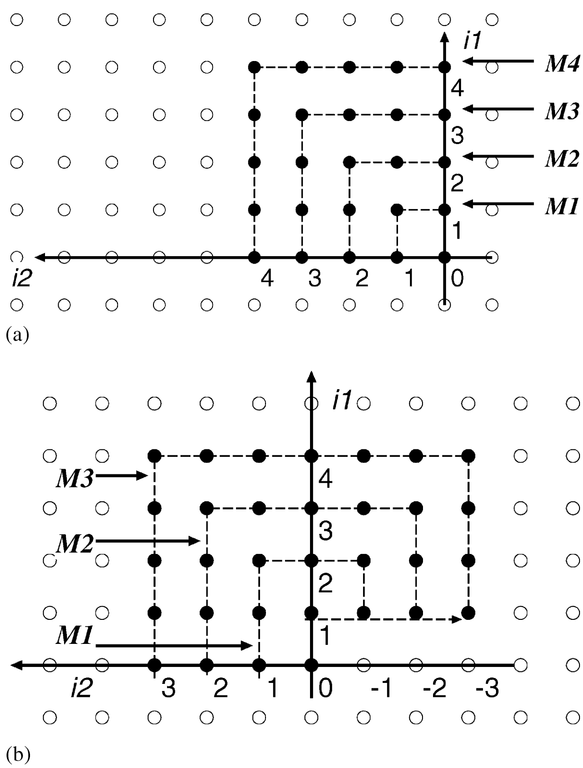

Fig. 4. Examples: (a) strongly causal square support region, (b) causal support region.

$\mathscr{M}_2$ is known. The estimation of the increased-order MSE filter corresponding to $\mathscr{M}_3$ is accomplished using left-hand side recursions following the updating scheme:

$$\mathscr{M}_2 \cup \{(0,3))\} \cup \{(1,3)\} \cup \{(2,3)\} \cup \{(3,0)\}\{(3,1)\} \cup \{(3,2)\} \cup \{(3,3)\}.$$

In the more general case, given the optimum 2-D filter, corresponding to a strongly causal support region, defined by the rectangular mask $\mathscr{M} = (0,0) \times (M, N)$, the order updating scheme producing the increased-order optimum filter, defined by the rectangular mask $\mathscr{M}^+ = (0,0) \times (M+1, N+1)$, is described in Table 3a.

The causal rectangular mask is similarly handled. The passage from the MSE filter corresponding $\mathscr{M}_1$ to the successive $\mathscr{M}_2$ is obtained using left- or right-hand side recursions as

$$\underbrace{\mathscr{M}_1 \cup \{(0,2)\} \cup \{(1,2)\} \cup \{(2,2)\} \cup \{(3,0)\} \cup \{(3,1)\} \cup \{(3,2)\}}_{\text{left-hand side recursions}},$$

$$\underbrace{\cup \{(1,-2)\} \cup \{(2,-2))\} \cup \{(3,-1)\} \cup \{(3,-2)\}}_{\text{right-hand side recursions}}.$$

Given the optimum 2-D filter, corresponding to a causal support region, defined by the mask $\mathscr{M} = (0,0) \times (M, N_1) \cup (1,0) \times (M, -N_2)$, the order updating scheme producing the increased-order optimum filter, defined by the mask $\mathscr{M}^+ = (0,0) \times (M+1, N_1+1) \cup (1,0) \times (M+1, -N_2-1)$, is illustrated in Table 3b.

The initialization of the recursive equations appearing in Tables 1 and 2 is obtained by direct solving of Eqs. (8) and (31) for $\mathscr{M} = \{(0,0)\}$.

The computational complexity of the algorithm summarized in Tables 1 and 2 is $O((l_1 - k_1 + 1)P)$ operations per recursion, where $P = \dim(\mathscr{C}_{\mathscr{M}}) = \sum_{i_1=k_1}^{l_1} (l_2(i_1) - k_2(i_1) + 1)$. Then, for a 2-D filter of a final mask shape $\mathscr{M}^{\text{fin}}$,

$$O((l_1^{\text{fin}} - k_1^{\text{fin}} + 1)(P^{\text{fin}})^2)$$

operations are required. For the special case of rectangular-shaped masks this amount equals to the complexity of the multichannel LWR algorithm. The nonrectangular mask case cannot be handled by the LWR algorithm, unless overparametrization is utilized [2].

Table 3a
Order updating recursions for a strongly causal support region

Given $\mathscr{C}_{\mathscr{M}}$
FOR $i_1 = 0{:}M$
Estimate $\mathscr{C}_{\mathscr{M}+L(i_1)}$ at position $(i_1, N+1)$
END
FOR $i_2 = 0{:}N+1$
Estimate $\mathscr{C}_{\mathscr{M}+L(i_2)}$ at position $(M+1, i_2)$
END

Table 3b
Order updating recursions for a causal support region

Given $\mathscr{C}_{\mathscr{M}}$
FOR $i_1 = 0{:}M$
Estimate $\mathscr{C}_{\mathscr{M}+L(i_1)}$ at position (i_1, N_1+1)
END
FOR $i_2 = 0{:}N_1+1$
Estimate $\mathscr{C}_{\mathscr{M}+L(i_2)}$ at position $(M+1, i_2)$
END
FOR $i_1 = 1{:}M+1$
Estimate $\mathscr{C}_{\mathscr{M}+R(i_1)}$ at position $(i_1, -N_2-1)$
END
FOR $i_2 = 0{:}-N_2-1$
Estimate $\mathscr{C}_{\mathscr{M}+R(i_2)}$ at position $(M+1, i_2)$
END

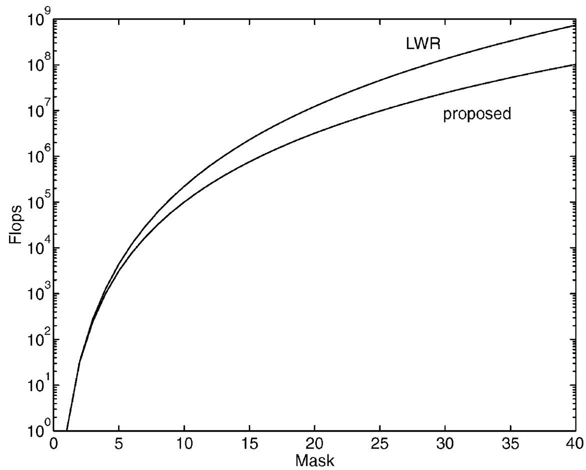

Fig. 5. Complexity count for the recursive estimation filters of strongly causal square support regions.

A great advantage that the proposed algorithm offers compared to its conventional LWR-based counterparts 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 $\mathscr{M} = (0,\ 0) \times (0, k^{\mathrm{fin}})$. When all filters of intermediate order $(0,\ 0) \times (k, k)$ are required, for all $1 \leqslant k \leqslant k^{\mathrm{fin}}$, algorithms of [17,18] require a repetitive application of LWR-based recursions, thus resulting in $\mathrm{O}(k^{\mathrm{fin}\,6})$ cost, which is outperformed by the $\mathrm{O}(k^{\mathrm{fin}\,5})$ cost of the proposed method. A comparison between the computational cost required by the LWR algorithm and the proposed one, for the recursive estimation of square masks as described by the first example above, is given in Fig. 5.

Thus, it is established that in all cases the proposed highly efficient order recursive 2-D algorithm has reduced computational complexity of any known scheme.

3.6. Structure estimation

The structure of a finite support 2-D FIR system is determined by the shape of the 2-D filter mask $\mathscr{M}$. Since our problem is formulated as a linear regression, structure determination can be tackled by exhaustive search procedures. Standard order selection criteria, such as the AIC, and the BIC, or more recently proposed methods [4,13], can be applied for structure determination of the 2-D model. The order recursive algorithm proposed in this paper can be used as a computational engine for the structure determination of hierarchically structured 2-D FIR and 2-D AR models. The algorithm provides recursions for the order update of the 2-D filter, by adding a single filter coefficient at each step. The proposed scheme updates the structure of the model, either in a predetermined way, or dynamically according to a specific order selection criterion.

All models up to a specified structured indexed by $\mathscr{M}_{\max}$ are estimated. Then, order estimation indices can be utilized for the location of the best model, that combines a minimum fitting error, with the lowest number of parameters. The AIC_w and the BIC are two popular criteria that can directly be applied based on the minimum squared error $E(N, \mathscr{M})$ and the number of processed data $N = N_{i1} \times N_{i2}$, They have the form

$$\mathrm{AIC}_w(N, \mathscr{M}) = N \ln(E(N, \mathscr{M})/(N - P)) + wP, \quad w > 0,$$

$$\mathrm{BIC}(N, \mathscr{M}) = N \ln(E(N, \mathscr{M})/(N - P)) + P \ln(N),$$

where P is the number of parameters associated with the filter mask $\mathscr{M}$. $E(N,\mathscr{M})$ is the MSE error attained, i.e.,

$$E(N,\mathscr{M}) = \mathscr{E}(z(n_1,n_2)) + \mathscr{D}^{\mathrm{t}}_{\mathscr{M}}\mathscr{C}_{\mathscr{M}}.$$

4. Simulation

The proposed algorithm is illustrated by an example from 2-D spectral parametric estimation. The task is to fit a 2-D autoregressive (AR) model to a given data set and then estimate the power spectral density from the model parameters. Indeed, let $x(n_1,n_2)$ be the output of a 2-D AR model driven by the 2-D white noise sequence $w(i_1,i_2)$ [18],

$$x(n_1,n_2) = -\sum_{(i_1,i_2)\in\mathscr{M}-\{0,\ 0\}} a_{i_1,i_2}x(n_1-i_1,n_2-i_2) + w(i_1,i_2). \tag{73}$$

The MSE 2-D AR model parameters, as well as the minimum prediction error attained, are obtained by the solution of the normal equations (6). As observations of the 2-D signal $x(n_1,n_2)$ are only available on a finite set, $[0,\ 0]\times[N_1,N_2]$, autocorrelation lags are estimated as

$$\hat{\rho}(i_1,i_2) = \frac{1}{(1+N_1)(1+N_2)}\sum_{n_1=0}^{N_1-i_1}\sum_{n_2=0}^{N_2-i_1} x(n_1+i_1,n_2+i_2)x(i_1,i_2). \tag{74}$$

Only causal support regions $\mathscr{M}$ for the AR model of Eq. (73) will be considered here. The general noncausal case can be handled similarly. Then, the power spectral density of $x(i_1,i_2)$ is estimated in terms of model parameters a_{i_1,i_2} as

$$P_{\mathrm{AR}}(f_1,f_2) = \frac{T_1T_2\sigma_w}{|1+\sum_{(i_1,i_2)\in\mathscr{M}-\{0,\ 0\}}a_{i_1,i_2}\mathrm{e}^{-j2\pi(f_1i_1T_1+f_2i_2T_2)}|}, \tag{75}$$

where T_1 and T_2 are the sampling period on the i_1 and i_2 space directions, respectively.

In our case, the 2-D data were generated by the superposition of two 2-D sinusoids as

$$x(i_1,i_2) = \sin(2\pi f_1(1)i_1 + 2\pi f_2(1)i_2)\sin(2\pi f_1(2)i_1 + 2\pi f_2(2)i_2) + w(i_1,i_2).$$

Frequencies are chosen to be $(f_1(1),f_2(1)) = (150,150)$ and $(f_1(2),f_2(2)) = (200,200)$. $w(i_1,i_2)$ is a 2-D white noise. The observation window is fixed to $[0,\ 0]\times[128,128]$. The support region $\mathscr{M}$ is considered to be a square mask. The algorithm of Table 1 is applied for the solution of the normal equations (6). This time, the left-hand side recursions are only utilized. The order recursions are organized as is discussed in the first example of the previous section. The estimated power spectral density for two different SNR (signal-to-noise ratio) is depicted in Figs. 6 and 7. The support region in both cases was the square mask $\mathscr{M} = [0,\ 0]\times[5,5]$.

The performance of the algorithm as a search engine in the 2-D structure estimation is illustrated next. Let us consider an 2-D FIR filter of the form

$$x(n_1,n_2) = -\sum_{i_1=0}^{3}\sum_{i_2=0}^{3} c_{i_1,i_2}x(n_1-i_1,n_2-i_2) + w(n_1,n_2), \tag{76}$$

where $x(n_1,n_2)$ is the input signal, and $w(n_1,n_2)$ is the disturbance signal. $x(n_1,n_2)$ are $w(n_1,n_2)$ white noise, zero mean, uncorrelated sequences. Clearly, the filter described by Eq. (76) has a strongly causal support region, and corresponds to the filter mask $\mathscr{M}$, of Fig. 4. The observation window is fixed to $[0,\ 0]\times[128,128]$. The SNR level is set to 30 dB.

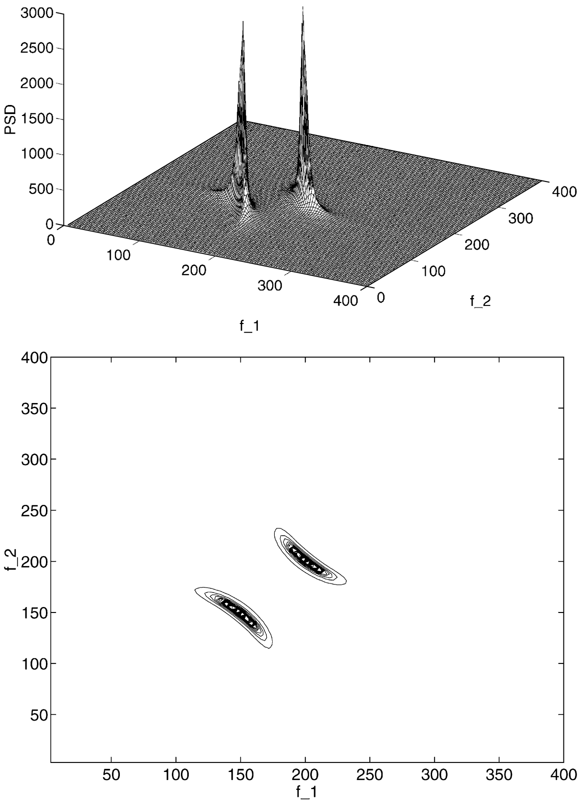

Fig. 6. Power spectral density of two 2-D sinusoids in SNR = 20 dB.

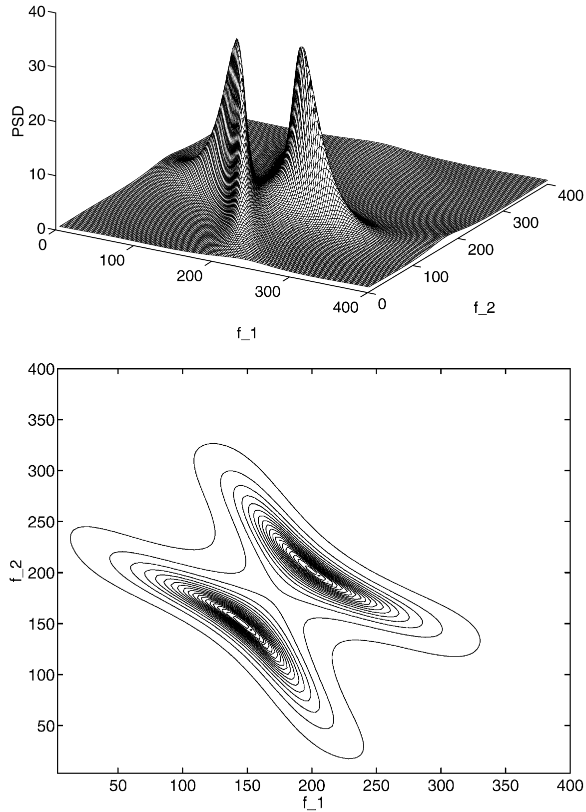

Fig. 7. Power spectral density of two 2-D sinusoids in SNR = 2 dB.

The proposed algorithm is applied for recursive estimation of filter structures coresponding to support regions $\mathscr{M}_1$, $\mathscr{M}_2$, up to $\mathscr{M}_5$, following Fig. 4a. In this case, recursions of Table 1 are only utilized. The algorithm estimates recursively a sequence of neighboring filters, by adding a single element at a time, from the starting point (0, 0) up to the point (5, 5), as

$$\underbrace{(0,\ 0),\ (0,\ 1),\ (1,\ 0),\ (1,\ 1)}_{\mathscr{M}_1},$$

$$\underbrace{(0,2),\ (1,2),\ (2,0),\ (2,1),\ (2,2)}_{\mathscr{M}_2},$$

$$\underbrace{(0,3),\ (1,3),\ (2,3),\ (3,0),\ (3,1),\ (3,2),\ (3,3)}_{\mathscr{M}_3},$$

$$\underbrace{(0,4),\ (1,4),\ (2,4),\ (3,4),\ (4,0),\ (4,1),\ (4,2),\ (4,3),\ (4,4)}_{\mathscr{M}_4},$$

$$\underbrace{(0,5),\ (1,5),\ (2,5),\ (3,5),\ (4,5),\ (5,0),\ (5,1),\ (5,2),\ (5,3),\ (5,4),\ (5,5)}_{\mathscr{M}_5}.$$

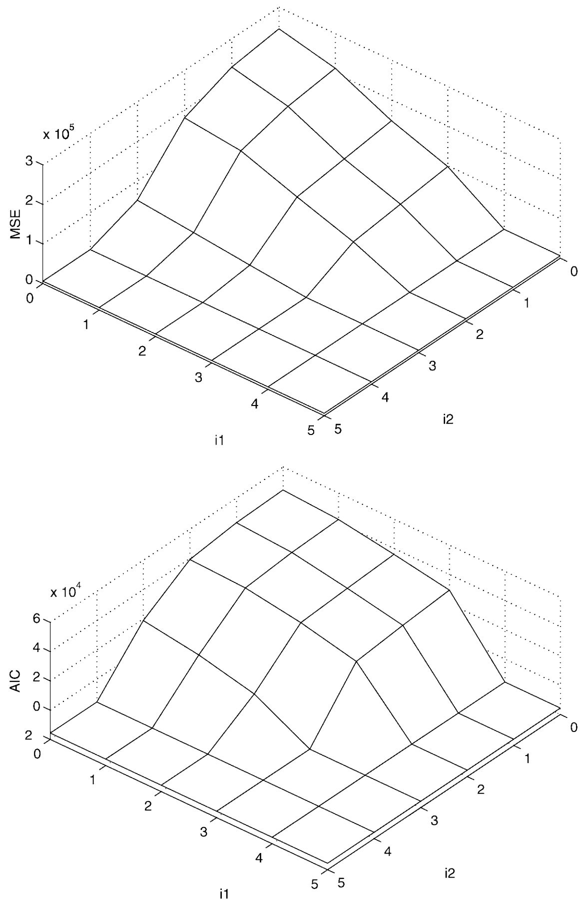

Fig. 8. MSE and AIC for a strongly causal rectangular structure estimation example.

The AIC_2 is applied based for the order estimation of the 2-D filter structure

$$AIC_2(N, \mathscr{M}) = N \ln(E(N, \mathscr{M})/(N - P)) + 2P.$$

The results obtained are depicted in Fig. 8. Fig. 8a illustrates the MSE error attained, while Fig. 8b shows the corresponding AIC_2 values. The minimum AIC_2 is attained at point (3, 3) which corresponds to the real filter structure.

5. 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. Matlab Code implementing the algorithm is available to interested readers.

References

[1] B. Aksasse, L. Badidi, L. Radouane, Two-dimensional autoregressive (2-D AR) model order estimation, IEEE Trans. Signal Process. 47 (July 1999) 2072–2077.

[2] Y. Boutalis, S. Kollias, G. Carayannis, A fast multichannel approach to adaptive image estimation, IEEE Trans. Acoust. Speech Signal Process. ASSP-37 (1989) 1090–1098.

[3] B. Choi, An order-recursive algorithm to solve the 3-D Yule–Walker equations of causal 3-D AR models, IEEE Trans. Signal Process. 47 (September 1999) 2491–2502.

[4] P. Djuric, S. Kay, Model order estimation of 2D autoregressive processes, IEEE International Conference on Acoustics, Speech and Signal Processing, 1991, 3405–3408.

[5] G. Glentis, N. Kalouptsidis, Fast algorithms for 2-D least squares FIR filtering, IEEE International Conference on Acoustics, Speech Signal Process., 1991.

[6] G. Glentis, N. Kalouptsidis, Efficient order recursive algorithms for multichannel least squares filtering, IEEE Trans. Signal Process. 40 (June 1992) 1354–1374.

[7] G. Glentis, N. Kalouptsidis, Efficient multichannel FIR filtering using a step versatile order recursive algorithm, Signal Process. 37 (1994) 437–462.

[8] G. Glentis, C. Slump, O. Herrmann, An efficient algorithm for two-dimensional FIR filtering and system identification, SPIE VCIP-94, Chicago, 1994, 220–232.

[9] G. Glentis, C. Slump, O. Herrmann, A versatile algorithm for two-dimensional symmetric noncausal modeling, IEEE International Conference on Image Processing-94, Austin, 1994.

[10] G. Glentis, C. Slump, O. Herrmann, A versatile algorithm for two-dimensional symmetric noncausal modeling, IEEE Trans. Circuits Systems II 45, (99. 251–256. (February 1998).

[11] B.F. McGuffin, B. Liu, An efficient algorithm for two-dimensional autoregressive spectrum estimation, IEEE Trans. Acoust. Speech Signal Process. ASSP-37 (1989) 106–117.

[12] A.K. Jain, Fundamentals of Digital Image Processing, Prentice-Hall, Englewood Cliffs, NJ, 1989.

[13] R. Kashyap, R. Chellappa, Estimation and choice of neighbors in spatial-interaction models of images, IEEE Trans. Inform. Theory IT-29 (January 1983) 60–72.

[14] H. Kaufman, A.M. Tekalp, Survey of estimation techniques in image restoration, IEEE Control Systems 1 (11) (January 1991) 16–23.

[15] A.H. Kayran, U. Kucuk, S.P. Parker, Two-dimensional Schur algorithm, Multidimentional Systems Signal Process. 9 (1) (January 1998) 7–37.

[16] J.V. Krogmeier, K.S. Arun, On the recursive computation of interpolators with nonrectangular masks, IEEE Trans. Signal Process. 44 (May 1996) 1072–1079.

[17] D. Manolakis, V. Ingle, Fast algorithms for 2-D FIR Wiener filtering and linear prediction, IEEE Trans. Circuits Systems CAS-34 (February 1987) 181–185.

[18] S.L. Marple, Digital Spectral Analysis with Applications, Prentice-Hall, Englewood Cliffs, NJ, 1987.

[19] R. Rajagopalan, M. Orchard, K. Ramchandran, Optimal supports for linear predictive models, IEEE International Conference on Image Processing, Austin 1994, pp. 785–789.

[20] R. Rajagopalan, M. Orchard, K. Ramchandran, Optimal supports for linear predictive models, IEEE Trans. Signal Process. 44 (December 1996) 3150–3154.