Hindawi Publishing Corporation
EURASIP Journal on Advances in Signal Processing
Volume 2008, Article ID 816507, 13 pages
doi:10.1155/2008/816507

Research Article

Sparse Deconvolution Using Support Vector Machines

José Luis Rojo-Álvarez,[1] Manel Martínez-Ramón,[2] Jordi Muñoz-Marí,[3] Gustavo Camps-Valls,[3] Carlos M. Cruz,[2] and Aníbal R. Figueiras-Vidal[2]

[1] *Department of Signal Theory and Communications, Universidad Rey Juan Carlos, Camino del Molino s/n, 28943 Fuenlabrada, Madrid, Spain*
[2] *Department of Signal Theory and Communications, Universidad Carlos III de Madrid, Avda Universidad, 30, 28911 Leganés, Madrid, Spain*
[3] *Departament d'Enginyeria Electrònica, Universitat de València, 46010 València, Spain*

Correspondence should be addressed to José Luis Rojo-Álvarez, joseluis.rojo@urjc.es

Received 26 September 2007; Revised 18 January 2008; Accepted 17 March 2008

Recommended by Theodoros Evgeniou

Sparse deconvolution is a classical subject in digital signal processing, having many practical applications. Support vector machine (SVM) algorithms show a series of characteristics, such as sparse solutions and implicit regularization, which make them attractive for solving sparse deconvolution problems. Here, a sparse deconvolution algorithm based on the SVM framework for signal processing is presented and analyzed, including comparative evaluations of its performance from the points of view of estimation and detection capabilities, and of robustness with respect to non-Gaussian additive noise.

1. INTRODUCTION

The sparse deconvolution (SD) problem consists in the estimation of an unknown sparse sequence which has been convolved with a (known) time series (impulse response of the system or source wavelet) and corrupted by noise, producing the observed noisy time series. The nonnull samples of the sparse series contain relevant information about the underlying physical phenomenon in each application. After its first application to reflective seismology [1], SD has been also used in a number of fields, such as ultrasonic nondestructive evaluation, image restoration, or cancelation of fractionated activity in intracardiac electrograms [2–4].

There has been an intensive development of SD techniques and algorithms. Since they are easy to compute, least squares (LSs) methods [5] have been used to deconvolve sparse seismic signals [6–9]. Some of the drawbacks of LS methods are their lack of robustness (they are very sensitive to non-Gaussian noise), and the ill-posing of the problem when the number of unknowns is close to the number of observations. Many complementary methods have been presented, such as Tikhonov's regularization [10] (also known in the statistical literature as ridge regression [11]), total LS [12], covariance shaping LS estimation [13], or signal-adaptive constraints [14]. The L_1 loss solution to SD [7] offers advantage in terms of sparseness when compared to the L_2 loss solution, also known as mean squared error (MSE) criterion technique. L_1-penalized SD was proposed later [15], by including an additional regularization term which consists on the L_1-norm of the unknown signal, allowing to control the sparseness of the solution. The Simplex algorithm was introduced as an optimization method for L_1-penalized SD in [2]. More recently, large scale linear programming methods have been also presented [16].

Maximum-likelihood deconvolution (MLD) was first developed by using state-variable models [17], but a more widespread used convolutional model was presented in [1], assuming that the sparse signal has a Bernoulli-Gaussian (BG) distribution and that the impulse response can be modeled as an autoregressive and moving average (ARMA) signal model. The Gaussian mixture (GM) model for SD assumes that the prior distribution of the sparse signal is a mixture of a narrow and a broad Gaussian distributions [18], and the BG distribution is a special case of the GM. Based on an extensive work in robust estimation [13, 19–21],

a minimax MSE estimation filter that minimizes some measure of the worst case pointwise MSE was developed in [22]. In [23], a double postprocessing was applied after an inverse filtering of the observations, which corresponds to a regularization in the Tikhonov sense and a subsequent noise attenuation.

In general, the performance of SD algorithms can be degraded by several causes. First, they can result in numerically ill-posed problems [18], and regularization methods are often required. Also when the knowledge about the noise is limited, both LS deconvolution (suitable for a Gaussian noise) and MLD can yield suboptimal solutions [24]. Finally, if a nonminimum phase (known) sequence is used, some SD algorithms can fail because the required inverse filtering turns unstable.

During the last years, support vector machine (SVM) algorithms have been shown to serve as a powerful framework for many data processing problems, with emphasis for classification and regression [25], and currently they constitute a fundamental tool in different relevant application domains [26]. SVM algorithms make use of several powerful concepts, like the structural risk minimization (SRM) principle, and they permit readily solving nonlinear problems while avoiding local minima. Some of the main advantages of SVM algorithms are (1) their solution is expressed in terms of a subset of the observations, the so-called support vectors (SVs), hence being intrinsically a sparse solution; (2) they are regularized in a smart and highly efficient way, as they follow the SRM principle; (3) they show very good performance in ill-posed problems (high-dimensional input features, low number of training samples); (4) nonlinear versions of SV algorithms can be readily obtained by means of the kernel trick, thus allowing a simple geometrical interpretation [27]; and (5) the ε -Huber cost, which is robust against impulse noise [28–30], can be used.

Though many of the above advantages are related to SD algorithms requirements, no attempt to develop an SV algorithm for SD has been done so far. In this paper, we propose an SVM-based SD algorithm, by conveniently exploiting the properties of the SVM methodology. Our approach follows two consecutive steps. First, we follow the linear signal processing framework presented in [28] to create a robust, linear SVM algorithm for SD. This approach, called primal signal model (PSM) algorithm, does not ensure sparseness in the estimated unknown signal, but it highlights the implicit relationship between the impulse response and its autocorrelation with Mercer's kernels that are used in SVM algorithms. Then based on a dual signal model (DSM) formulation, similar to that previously proposed for interpolation of time series [31], we present an SD algorithm which uses an equivalent problem statement to obtain a sparse solution. Given that the impulse response of a deconvolution problem will not be always a valid Mercer kernel, a filtered version of the observations is used, in such a way that the autocorrelation of the impulse response can be used as Mercer's kernel in the equivalent problem. We call this practical algorithm autocorrelation kernel signal model (AKSM).

The rest of the paper is organized as follows. In Section 2, the equations for the SD problem are introduced, together with a brief description of the two representative SD algorithms from the literature that will be used for benchmarking. In Section 3, preliminary elements that are required for the formulation of the practical SVM algorithm for SD are summarized. In Section 4, the AKSM algorithm is developed and analyzed in detail. In Section 5, we present experiments showing the performance of the AKSM procedure. Finally, some conclusions are drawn.

2. PROBLEM STATEMENT

Be $y[n]$ a discrete-time signal which contains in its lags $0 \leq n \leq N$ a set of $N+1$ observed samples of a time series that is assumed to be the result of the convolution between an unknown sparse signal $x[n]$, whose samples in the lags $0 \leq n \leq M$ we want to estimate, and a time series $h[n]$, whose samples in the lags $0 \leq n \leq Q-1$ are known. Samples of $y[n]$, $x[n]$, and $h[n]$ are equal to zero outside the defined limits, and also, $N = M + Q + 1$. Then the following analysis model can be stated:

$$y[n] = \hat{x}[n] * h[n] + e[n] = \sum_{j=0}^{M} \hat{x}[j] h[n-j] + e[n], \quad (1)$$

where $*$ denotes the discrete-time convolution operator; $\hat{x}[n]$ is the estimation of the unknown input signal, and $e[n]$ is the estimation error accounting for the model residuals. Equation (1) is required to be fulfilled for lags $n = 0, \ldots, N$ of observed signal $y[n]$.

The solution is usually regularized [10] by minimizing the qth power of the q-norm of estimate $\hat{x}[n]$, hence forcing smooth solutions. The functional to be minimized should take into account both this q-norm term and an empirical error measurement term defined according to an a priori determined pth power of the p-norm of residuals $e[n]$. Hence the regularized functional to be minimized can be written as

$$J_D = \sum_{n=0}^{N} ||e[n]||_p^p + \lambda \sum_{n=0}^{N} ||\hat{x}[n]||_q^q, \quad (2)$$

where parameter λ tunes the tradeoff between model complexity and the minimization of estimation errors. It is worth noting that different norms can be adopted, involving different families of models and solutions. For instance, setting $p = 2$, $\lambda = 0$ yields the LS criterion, whereas setting $p = 2$, $\lambda \neq 0$, $q = 2$ yields the well-known Tikhonov-Miller regularized criterion [10]; for $p = 1$, $\lambda = 0$, we obtain the L_1 criterion, and setting $p = 1$, $\lambda \neq 0$, $q = 1$ provides the L_1-penalized method [2].

From the point of view of MLD [1, 17] or GM [18], SD algorithms can be written as the minimization of the (power of the) L_p norm of the residuals minus a regularization term which consists of the log-likelihood of the sparse time series for an appropriate statistical model, that is,

$$J_{\mathrm{MLD}} = \sum_{n=0}^{N} ||e[n]||_p^p - \lambda \sum_{n=0}^{N} \log l(\hat{x}[n]), \quad (3)$$

where $l(\hat{x}[n])$ denotes the likelihood, according to the assumptions about the input signal statistics. White Laplacian statistics lead to the L_1-penalized criterion. Alternatively, non-Gaussian input signal statistics have been proposed, such as BG and GM distributions [1, 18]. Actually, the BG distribution can be seen as a special case of a GM optimization. SD with a GM assumes that the prior distribution of the sparse signal can be approximated with a mixture of two zero-mean Gaussians, one of them being narrow (small variance) and the other being broad (larger variance) [18]. Therefore, the functional to be minimized in GM deconvolution is

$$J_{\mathrm{GM}} = \sum_{n=0}^{N} ||e[n]||^2 - \lambda_{\mathrm{gm}} \sum_{n=0}^{N} \log \sum_{j=1}^{2} \pi_j p_j(\hat{x}[n]), \tag{4}$$

where π_j are the prior probabilities of each Gaussian component, and

$$p_j(\hat{x}[n]) = \left(\sigma_j \sqrt{2\pi}\right)^{-1} \exp\left(-\frac{\hat{x}[n]^2}{2\sigma_j^2}\right) \tag{5}$$

are their probability densities.

These functionals contain a term that is intended to minimize the training error of the solution plus a term that can be viewed as a regularization term from the point of view of Tikhonov. This is a term that controls the overfiting of the obtained solution.

This term, often called stabilizer, can also be viewed as a control of the smoothness of the solution. In other words, as the function that we want to approximate must be smooth, a good criterion consists of stablishing a tradeoff between the minimization of an error term and the maximization of the smoothness. As it has been stated in [32, 33], the stabilizers used in equations to are smoothers as lower values of these functions produce smoother solutions. We will use here an SVM functional, which also contains a regularization term that produces smooth solutions.

3. SVM FOR SD: PRELIMINARY CONSIDERATIONS

In this section, an algorithm for SD problems using a SVM formulation is presented. For this purpose, several basic elements of SVM algorithms are first introduced. Then, a SVM formulation for SD is initially proposed by following the SVM digital signal processing framework presented in [28]. The purpose of this approach is to highlight the signal structure which is implicit in the algorithm, in particular, the autocorrelation of the impulse response. All elements that are considered in this section will be used in Section 4 to introduce the proposed algorithm.

3.1. Preliminary SVM elements for signal processing

3.1.1. Mercer's kernels

Let us consider the nonlinear regression paradigm presented in [31], in which a set of observations $y[n]$ is modeled as a nonlinear regression on time instants n. This signal model uses a nonlinear transformation $\boldsymbol{\phi} : t \in \mathbb{R} \rightarrow \mathcal{H}$, which maps a real scalar to a *feature space* $\mathcal{H}$. For a properly chosen transformation $\boldsymbol{\phi}$, a linear regression model can be built in $\mathcal{H}$. This can be done through a theorem provided by Mercer [34] in the early 1900s. Mercer's theorem shows that there exist a function $\boldsymbol{\phi} : \mathbf{x} \in \mathbb{R}^n \rightarrow \mathcal{H}$ and a dot product:

$$K(\mathbf{x}_i, \mathbf{x}_k) = \boldsymbol{\phi}(\mathbf{x}_i)^T \boldsymbol{\phi}(\mathbf{x}_k), \tag{6}$$

if and only if $K(\cdot, \cdot)$ is a positive integral operator on a Hilbert space, that is, if and only if for any function $g(\mathbf{x})$ for which

$$\int g(\mathbf{x}) d\mathbf{x} < \infty, \tag{7}$$

the inequality

$$\int K(\mathbf{x}, \mathbf{y}) g(\mathbf{x}) g(\mathbf{y}) d\mathbf{x}\, d\mathbf{y} \geq 0 \tag{8}$$

holds. Hilbert spaces provided with dot products that fit Mercer's theorem are often called reproducing kernel Hilbert space (RKHS).

We use here a nonlinear transformation whose domain is constrained to discrete time lags, $\boldsymbol{\phi} : n \in (Z) \in \mathbb{R} \rightarrow \mathcal{H}$, that is, given that we are dealing with discrete time processes, this mapping is particularized to integers, in order to fit later into the deconvolution model equation. For this nonlinear transformation $\boldsymbol{\phi}$, a linear regression model into the Hilbert space will be given by

$$y[n] = \boldsymbol{v}^T \boldsymbol{\phi}[n] + e[n], \tag{9}$$

where $\boldsymbol{v} \in \mathcal{H}$ is the vector with the regression weights and $e[n]$ are the residuals. Provided that $\boldsymbol{v}$ lies in the subspace spanned by a set of vectors $\boldsymbol{\phi}[n]$, with $n = 0, \ldots, N$, one can express this vector as

$$\boldsymbol{v} = \sum_{m=0}^{N} \eta_m \boldsymbol{\phi}[m], \tag{10}$$

where $\eta_m \in \mathbb{R}$ are the coefficients of the vector expansion in that subspace, and then, the linear regression model can be expressed as

$$y[n] = \sum_{m=0}^{N} \eta_m \boldsymbol{\phi}[m]^T \boldsymbol{\phi}[n] + e[n] = \sum_{m=0}^{N} \eta_m K[m, n] + e[n]. \tag{11}$$

Therefore, we can use kernel function $K(m, n)$ to compute the dot products. For instance, in the widely used Gaussian Mercer's kernel, which is given by $K[m, n] = \exp(-(m-n)^2/2\sigma^2)$, the explicit expression of nonlinear transformation $\boldsymbol{\phi}[\cdot]$ is unknown, and the RKHS dimension is infinite [35].

3.1.2. Shift invariant and autocorrelation kernels

A particular class of kernels are shift invariant kernels, which are those fulfilling $K[m, n] = K[m - n]$. A necessary and

sufficient condition for shift invariant kernels to be Mercer's kernels [36] is that their Fourier transform be non-negative.

Let $s[n]$ be a discrete time, real signal with $s[n] = 0\ \forall n \notin [0, S-1]$, and let $R_s[n] = s[n]*s[-n]$ be its autocorrelation. Then the following kernel can be built:

$$K_s[m,n] = R_s[m-n], \tag{12}$$

which is called the autocorrelation-induced kernel (or just *autocorrelation kernel*) of signal $s[n]$. It is well known that the spectrum of an autocorrelation sequence is nonnegative, and given that (12) is also a shift-invariant kernel, it will be always a valid Mercer's kernel.

3.1.3. Residual cost function

The SVM algorithms for SD that will be presented share a common form for the driving criterion, which consists on minimizing

$$J_{\text{SVM}} = \sum_{n=0}^{M} L(e[n]) + ||\boldsymbol{\tau}||_2^2, \tag{13}$$

where $L(e[n])$ is a loss function of the residuals, and $\boldsymbol{\tau}$ is used to build the penalization term, and it will be defined for each SVM algorithm according to the signal model being used, as described later. Equation (13) is similar to (2) (identical if $\tau = \hat{\mathbf{x}}$, with $\hat{\mathbf{x}} = [\hat{x}[0], \ldots, \hat{x}[N]]^T$). For the purpose of SVM formulation, the robust ε-Huber cost function of the residuals [28] will be used here in (13), which will allow us to deal with different kinds of noise. Its form is

$$L_{\varepsilon H}(e[n]) = \begin{cases} 0, & |e[n]| \le \varepsilon \\ \dfrac{1}{2\gamma}(|e[n]| - \varepsilon)^2, & \varepsilon < |e[n]| \le e_C \\ C(|e[n]| - \varepsilon) - \dfrac{1}{2}\gamma C^2, & |e[n]| > e_C, \end{cases} \tag{14}$$

where ε, C, and γ are the free parameters of the cost function that have to be fixed according to some a priori knowledge of the problem. The ε-insensitive zone discards errors lower than ε, and, importantly, it allows to control the sparseness of the SVM solution; the quadratic cost zone is appropriate for a Gaussian noise; and finally, the linear cost zone is appropriate for an impulse noise [28, 30].

Note that conventional SVM regression algorithm uses Vapnik's ϵ-insensitive residual cost, which is basically a linear cost. However, this will not be adequate in the presence of Gaussian noise, which is a usual situation in discrete time signal processing. This fact was previously taken into account in the formulation of least-squares SVM [37], where a regularized quadratic cost is used for a variety of signal processing applications. But in this case, non-sparse solutions are obtained. The ε-Huber cost, proposed in [29, 38], is just a combination of both the quadratic and the ϵ-insensitive residual costs. In fact, it represents the ε-insensitive cost when $\gamma C \to 0$, Huber's robust cost when $\varepsilon = 0$ [30], and the quadratic cost when $\varepsilon = 0$, $\gamma C \to \infty$, thus generalizing all of them.

3.2. A signal model in the primal for deconvolution

Using the convolutional model in (1), it is possible to build an SVM algorithm that provides a robust estimation of the unknowns. This PSM algorithm can be formulated as the minimization of primal functional (13) when $\boldsymbol{\tau} = \hat{\mathbf{x}}$, that is, we minimize

$$J_{\text{PSM}} = \sum_{n=0}^{N} L_{\varepsilon H}(e[n]) + \frac{1}{2}||\hat{\mathbf{x}}||_2^2, \tag{15}$$

constrained to (1).

Given that the derivation of this SVM algorithm uses the convolutional signal model in (1), we will call it primal signal model (PSM) algorithm. The optimization of (15) is a constrained problem, which can be expressed into its dual form by using the Lagrange theorem and the corresponding Karush-Khun-Tucker (KKT) conditions. Appendix A contains a brief derivation of this SVM algorithm, and here we only stress the relevance of two sets of the KKT conditions, as well as their interpretation in terms of signal processing blocks, as this will be the basis of the proposed SD algorithm in Section 4.

First, one of the KKT conditions yields the unknown sparse signal being given by

$$\hat{x}[n] = \sum_{i=0}^{N} (\alpha_i - \alpha_i^*) h[i-n] = \sum_{i=0}^{N} \eta_i h[i-n], \tag{16}$$

where α_n, α_n^* are the Lagrange multipliers accounting for positive and negative residuals in (1), respectively, and each of their pairs can be grouped into a single model coefficient $\eta_i \in \mathbb{R}$, $i = 1, \ldots, N$. Second, a well-known relationship between model residuals $e[n]$ and the corresponding model coefficient η_n is

$$\eta_n = \left.\frac{\partial L_{\varepsilon H}(e)}{\partial e}\right|_{e=e[n]} = L'_{\varepsilon H}(e[n]), \tag{17}$$

which can be also shown by using the appropriate set of KKT conditions [28], and states that the model coefficients are a piecewise linear, yet simple function of the model residuals.

We can define a discrete time signal containing the model coefficients, as $\eta[n] = \eta_n$, for $n = 0, \ldots, N$, and being zero otherwise. Then we can express KKT conditions in (16) and (17) as follows:

$$\begin{aligned} \hat{x}[n] &= \eta[n]*h[-n]*\delta[n+Q], \\ \eta[n] &= L'_{\varepsilon H}(e[n]), \end{aligned} \tag{18}$$

respectively, where $\delta[n]$ is the Kronecker delta sequence (1 for $n = 0$ and zero elsewhere). On the one hand, the estimated sparse signal is expressed in terms of a linear, time-invariant system, as the convolution between the model coefficient signal and the shifted, time-reversed impulse response of the system. On the other hand, the model coefficients and the residuals are related by a nonlinear, static system. Taking (1) into account, we can consider a joint system that contains the three previously expressed signals

and systems relations. Several facts can be observed from these equations. First, according to the scheme, estimated signal $\hat{x}[n]$ should not be expected to be sparse in general, because similarly to SVM regression algorithm, it is the sparseness of $\eta[n]$ that can be controlled with ε parameter, and there is a convolutional relationship between $\hat{x}[n]$ and $\eta[n]$ that will depend on the (modified) impulse response, which in general does not have to be sparse. Second, note that there is no Mercer's kernel appearing explicitly in this formulation, as it should be expected in an SVM approach but, instead, there is an implicitly present autocorrelation kernel given by

$$R_h[n] = h[n]*h[-n], \tag{19}$$

if we associate the two systems containing the original system impulse response and its reversed version. However, the solution signal is embedded between them, which precludes the convenient use of this autocorrelation kernel in this PSM formulation. Third, the role of delay system $\delta[n+Q]$ can be interpreted as just an index compensation that makes causal the global system.

In summary, it can be said that the PSM algorithm yields a regularized solution, in which an autocorrelation kernel is implicitly used, but it does not allow to control the sparseness of the estimated signal.

4. A SUITABLE SVM FOR SD

An SVM algorithm for SD can be proposed which overcomes the limitations of the PSM algorithm. As described next, this algorithm uses a convenient transformation of the observed signal, as well as a different signal model as a starting point, namely, the nonlinear regression model presented in Section 3. These two modifications will lead us to work with an alternative convolutional model in which the new impulse response is given by the autocorrelation of the impulse response in the original problem, hence allowing us to explicitly use Mercer's kernels that are autocorrelation kernels. Additionally, in the new signal model the sparse unknown sequence can be straightforwardly identified by the Lagrange multipliers, and its sparseness can be readily controlled by means of parameter ε in the SVM algorithm.

Provided that the impulse response of the convolutional signal model in (1) is not necessarily an autocorrelation sequence, we start by making a previous filtering of observations $y[n]$ as follows:

$$z[n] = y[n]*(h[-n]*\delta[n+Q]) = \hat{z}[n] + e'[n], \tag{20}$$

where $e'[n] = e[n]*(h[-n]*\delta[n+Q])$ are the modified residuals. Note that the signals that we use here for filtering, that is, a reversed impulse response filter and a Q samples time shift, are present in the PSM algorithm. Under these conditions,

$$\begin{aligned}\hat{z}[n] &= \hat{x}[n]*(h[n]*h[-n]*\delta[n+Q]) \\ &= \hat{x}[n]*R_h[n]*\delta[n+Q],\end{aligned} \tag{21}$$

and the transformed signal follows a signal model which is similar to the original one, that is, a convolution between the same unknown sparse signal and the autocorrelation of the impulse response.

We call (21) the autocorrelation kernel signal model. Instead of the convolutional relationship for the observations and the data in (1), we start by the nonlinear regression SVM formulation between modified observations $z[n]$ and time instants n given in (9), that is, the SVM nonlinear regression model for the modified observations is here stated as

$$z[n] = \boldsymbol{v}^T\boldsymbol{\phi}[n] + e'[n], \tag{22}$$

where $\boldsymbol{v}$ is the coefficient vector in the RKHS. Accordingly, the penalization term to be used in the generic SVM criterion in (13) is obtained by making $\boldsymbol{\tau} = \boldsymbol{v}$, and the SVM optimization functional is given in this case by

$$J_{\text{AKSM}} = \sum_{n=0}^{N} L_{\varepsilon H}(e'[n]) + \frac{1}{2}||\boldsymbol{v}||_2^2. \tag{23}$$

The complete derivation of the dual problem is not presented here, but only the relevant conditions to be taken into account are highlighted. Appendix B contains further details on the corresponding dual problem, and the interested reader is addressed to [31] for the case of interpolation of discrete time series, which has a formally similar SVM dual problem derivation. Note that despite the derivation of SVM algorithms for interpolation and for SD may be formally similar, these two digital signal processing procedures are quite different.

The nonlinear relationship between the residuals and the new-model coefficients still holds, that is,

$$\eta_n = L'_{\varepsilon H}(e'[n]), \tag{24}$$

where η_n are the new-model coefficients corresponding to the positive and negative residuals given by (22). This relationship can be easily shown by examining the KKT conditions (see, e.g., [31]). Then the following expansion for the weight vector $\boldsymbol{v}$ can be obtained from the corresponding KKT condition as follows:

$$\boldsymbol{v} = \sum_{n=0}^{N} \eta_n \boldsymbol{\phi}[n]. \tag{25}$$

We can avoid to work explicitly in the RKHS by introducing this equation into the nonlinear regression signal model in (22), which yields

$$\hat{z}_n = \sum_{i=0}^{N} \eta_i \boldsymbol{\phi}[i]^T \boldsymbol{\phi}[n] \sum_{i=0}^{N} \eta_i K[i,n] = \sum_{i=0}^{N} \eta_i R_h[i-n], \tag{26}$$

where we have replaced generic Mercer's kernel $K(i,n)$ by autocorrelation kernel $R_h[i-n]$ generated by impulse response $h[n]$.

Using again signal $\eta[n]$ as previously to account for the model coefficients, it is clear that (26) can be seen as the following convolutional model:

$$\hat{z}[n] = \eta[n]*R_h[n] = \hat{x}[n]*R_h[n], \tag{27}$$

where unknown sparse signal $\hat{x}[n]$ can be straightforwardly assigned to model coefficients in $\eta[n]$. Thus $\hat{x}[n]$ sparseness can be now controlled by just tuning ε to obtain a sparse set of model coefficients. If we consider a block diagram of the joint system, the effect of transforming the observations and using an appropriate dual signal model, instead of the primal signal model in the preceding algorithm, is the obtention of an autocorrelation kernel and the straightforward assignment of the sparse unknown sequence to the model coefficients.

In conclusion, we can build a model from modified observations $z[n]$, which is described as the convolution between the original unknown signal and the autocorrelation of the impulse response, allowing us to control sparseness and to use a Mercer's kernel, hence keeping two of the major advantages of the SVM algorithms.

5. SIMULATION EXAMPLES

We will evaluate the performance of the proposed AKSM algorithm and compare it with other previously proposed algorithms. We have made extensive simulations with synthetic signals, both deterministic and BG distributed. Given that the results are qualitatively similar, only the results from a representative case (a deterministic unknown signal) are reported here. Among the wide number of available SD techniques in the literature, we choose L_1-penalized (minimizing (II)) and GM (minimizing (II)) algorithms for comparing their performances with that of the proposed AKSM algorithm. Finally, a real-data example is included applying the above SD methods to ultrasonic data.

5.1. Experimental setup

A deterministic sparse signal with 128 samples and five nonzero values ($x_{20} = 8$, $x_{25} = 6.845$, $x_{47} = -5.4$, $x_{71} = 4$, and $x_{95} = -3.6$) [39] was used for simulations. Peaks at samples 20 and 25 are close in order to test the algorithm resolution, whereas peak x_{95} is small in order to test detection capabilities. The other time series consists on the first 15 samples of the impulse response of

$$H(z) = \frac{z - 0.6}{z^2 - 0.414z + 0.64}. \tag{28}$$

A noise sequence was added, with a variance corresponding to a signal-to-noise ratio (SNR) from 4 to 20 dB. Performance was studied for three different kinds of additive noise: Gaussian, Laplacian, and uniform.

The free parameters for the L_1-penalized and for the GM algorithms are λ and λ_{gm}, respectively, whereas the free parameters for the AKSM algorithm are initially $\{C, \gamma, \varepsilon\}$. Taking into account that parameter ε in the AKSM algorithm controls the sparseness of the solution, the impact of changing $\{C, \gamma\}$ in SVM algorithms was first analyzed, previous to the performance analysis, which allowed to fix their values to $C = 100$, $\gamma = 10^{-3}$ for all the subsequent performance experiments, as we will explain later. Note that parameters $\lambda, \alpha_{\text{gm}}$, and ε have different meanings and purposes, therefore, different ranges need to be explored for each of them in the performance analysis.

The merit figures were obtained by averaging the results of the algorithms in 100 realizations. The same set of 100 realizations was used for each SNR and for each tested tradeoff parameter, in order to analyze their impact on the solution.

Performance was evaluated in terms of both estimation and detection capabilities of the algorithms. On the one hand, estimation quality was quantified by using

$$\begin{aligned} \text{MSE} &= \frac{1}{N+1}\sum_{n=0}^{N}\left(\hat{x}[n] - x[n]\right)^2, \\ F &= \sum_{x[n]\neq 0}\left(\hat{x}[n] - x[n]\right)^2, \\ F^{\text{null}} &= \sum_{x[n]=0}\left(\hat{x}[n] - x[n]\right)^2 = \sum_{x[n]=0}\hat{x}^2[n], \end{aligned} \tag{29}$$

where MSE denotes mean-squared error, which measures the global estimation capabilities, and F (F^{null}) represents the particularized estimation capabilities for spikes (for null samples) in the true sparse signal [2]. On the other hand, detection capabilities were measured by using the sensitivity (Se) and the specificity (Sp) of spike detection, given by

$$\text{Se} = \frac{V_+}{V_- + F_-}, \qquad \text{Sp} = \frac{V_-}{V_- + F_+}, \tag{30}$$

where V (F) represents the number of true (false) detections, and subindex + (−) represents positive or detected spike (negative or null) samples. The presence of a peak in the signal is determined by finding either local positive maxima or negative minima with an amplitude threshold equal to 1/100 times the absolute maximum of the signal.

5.2. AKSM-free parameters

A highly relevant issue in SVM algorithms is often the requirement of determining appropriate values for all the SVM-free parameters, which is usually addressed either by theoretical considerations [40, 41] or by computationally expensive cross-validation search using a validation data set [25]. Therefore, we started our analysis by studying the effect of free parameters C and γ on the AKSM algorithm.

For different examples of SNR values, we explored a grid of values for parameters γ, C, and ε, given by 100 points in a rank for each parameter, and obtained the MSE with respect to the true sparse signal. Figure 1 shows two examples of MSE curves for SNR = 16 dB and 4 dB. In the first case (panels (a) and (b), SNR = 16 dB) the experimentally best value of $\varepsilon = 0.9$ was fixed, and each MSE curve was obtained by fixing the value of the other free parameter, and averaging MSE in 100 realizations. It can be seen that, in average, there is a wide range for both C and γ that can be considered as appropriate. A qualitatively similar result can be observed for SNR = 4 dB, panels (c) and (d), obtained with fixed $\varepsilon = 2.4$, which was shown to be the experimentally best value for this case.

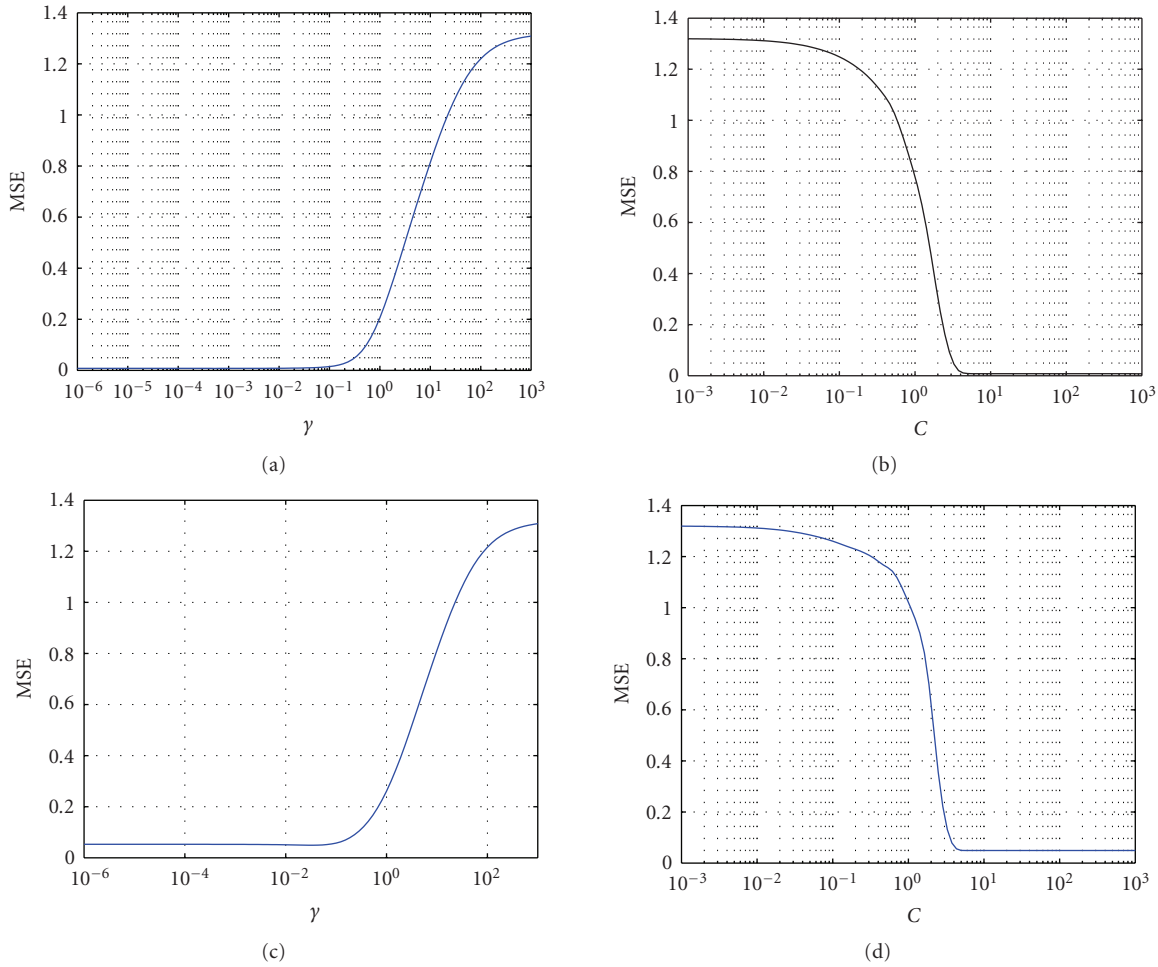

Figure 1: MSE of the AKSM algorithm as a function of free parameters γ and C in the ε-Huber cost function, with SNR = 16 dB (a,b) and SNR = 4 dB (c,d).

Therefore, aiming not to give SVM more information than to the other methods by means of its larger number of free parameters, $C = 10^2$ and $\gamma = 10^{-3}$ were fixed for all the simulations of the performance analysis.

5.3. Estimation performance

Figure 2 (panels (a), (b), and (c)) presents the MSE results for all the SD algorithms with different SNR, as a function of the free parameter of the algorithm, that is, λ for L_1-penalized, α_{gm} for GM, and ε for AKSM. It can be seen that the optimum value for the free parameter depends of SNR in all cases. The L_1-penalized and the AKSM algorithms show low performance with extremely low ($\lambda, \varepsilon \to 0$) and high ($\lambda, \varepsilon >$ 3) values of the free parameter, the optimum value being at some intermediate point. Lower values of the free parameter yielded less-sparse solutions for all algorithms. Higher values of the free parameter in L_1-penalized and AKSM lead to a constant value solution, clearly reached in L_1-penalized algorithm, called the *zero solution* [2], which corresponds to a null-estimated sparse signal, and hence a value of MSE proportional to the norm of the true sparse signal. The GM algorithm showed a stabilization for high values of the free parameter. The qualitatively different behavior of the GM algorithm for high values of α_{gm} is due to the fact that, in most cases, there was at least one detected peak. Hence the zero solution was not reached on average with this algorithm.

Table 1: Averaged MSE ± its standard deviation ($\times 10^2$) for optimally chosen free parameters for each SD algorithm (best in boldface, second in italic).

Method	4 dB	10 dB	16 dB	20 dB
L_1-pen.	*23.9±8.9**×	6.8 ± 2.9*	*1.5±0.6**×	*0.6±0.3**
GM	117.2 ± 197.6*	*5.2±6.5*	3.3 ± 2.0*	1.5 ± 0.3*
AKSM	**12.2 ± 4.5**	**3.2 ± 1.3**	**0.7 ± 0.2**	**0.3 ± 0.1**

*Denotes $p < .01$ for AKSM versus GM and for AKSM versus L1 comparisons.
×Denotes $p < .01$ for GM versus L1 comparison.

Mean and standard deviation of the MSE for each algorithm are summarized in Table 1, showing that the highest global estimation capabilities are always reached by the AKSM, followed by the L_1-penalized and the GM procedures. A Wilcoxon-paired test was made to check for

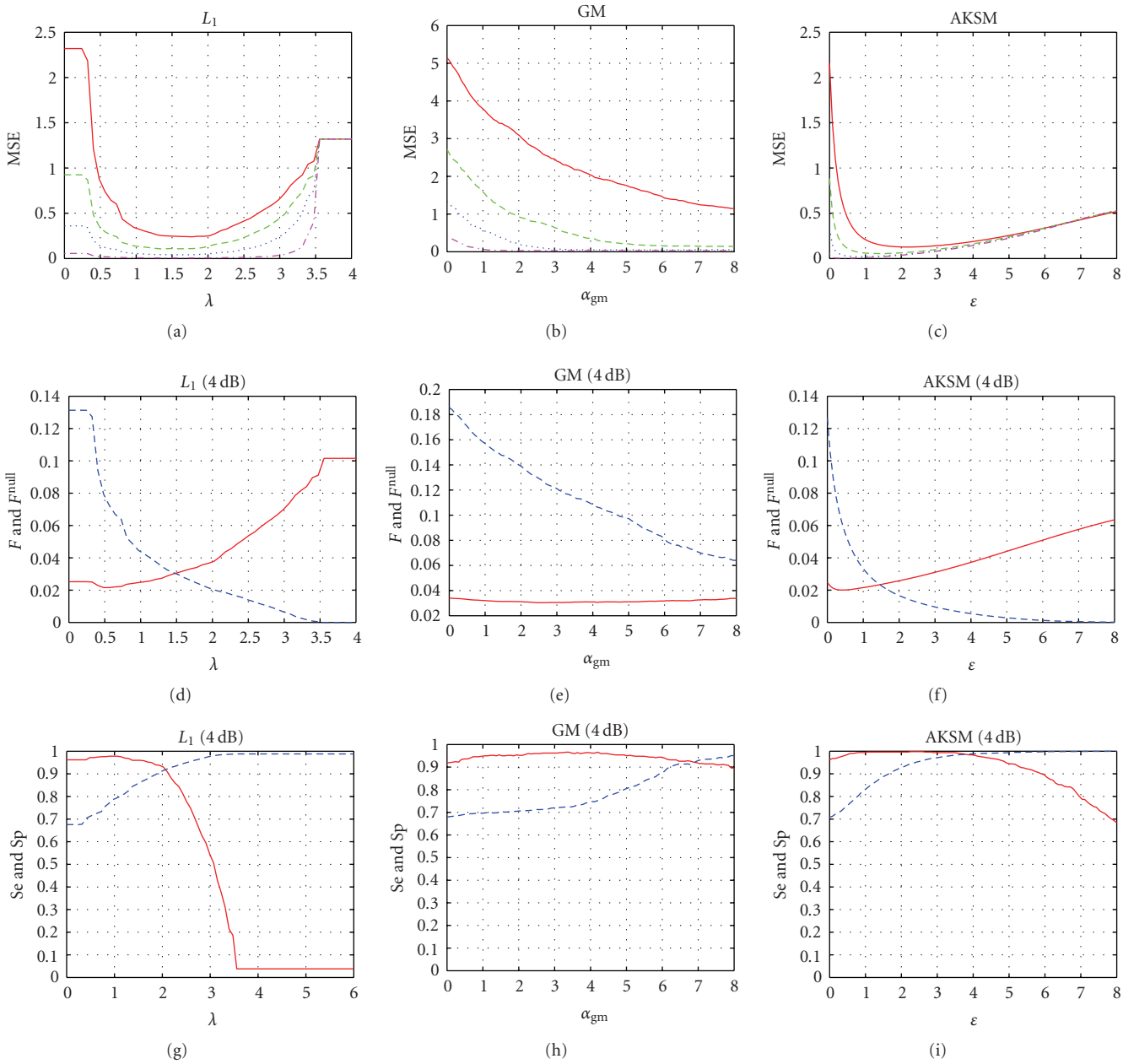

FIGURE 2: Performance analysis for the L_1-penalized algorithm (left), the GM algorithm (middle), and the AKSM algorithm (right). (a,b,c) estimation performance: averaged MSE for SNR = 4, 8, 12, and 20 dB (solid, dashed, dotted, and dash-dotted, resp.). (d,e,f) estimation performance: averaged F (solid) and F^{null} (dashed) with SNR = 4 dB. (g,h,i) detection performance: averaged Se (solid) and Sp (dashed) for SNR = 4 dB.

statistical significance in the observed differences ($*$ denotes $p < .01$ for AKSM versus GM and for AKSM versus $L1$ comparisons, $\times$ denotes $p < .01$ for GM versus $L1$ comparison).

In Figure 2 (panels (d), (e), and (f)), averaged F and F^{null} are depicted separately for each algorithm, for SNR = 4 dB. In general, it can be observed that the higher the free parameter, the higher F and the lower F^{null}. Also saturation effects are present in GM for F^{null}. The cross-point between both curves represents a good indicator of the tradeoff between both merit figures, and it appears for values of the free parameter close to the optimum in the MSE sense (i.e., for global estimation) in all the algorithms, independently of the SNR. This suggests that the global estimation performance was not biased towards neither of the two kinds of samples (nonnull and null) in the sparse signal.

5.4. Detection performance

Figure 2 (panels (g), (h), and (i)) shows the detection capabilities in terms of Se and Sp for all the algorithms. As expected, Sp (Se) increases (decreases) with the free

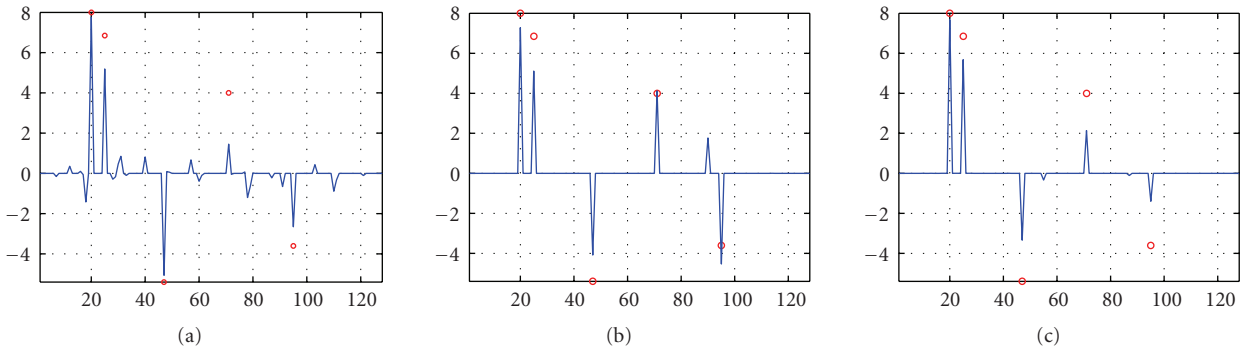

Figure 3: An example of SD results versus data with SNR = 4 dB for: (a) the L_1-penalized algorithm; (b) the GM algorithm; (c) the AKSM algorithm.

Table 2: Averaged MSE ± its standard deviation ($\times 10^2$) for each SD algorithm (best in bold, second in italic) under Laplacian and uniform additive noises.

	Laplacian noise				Uniform noise			
Method	4 dB	10 dB	16 dB	20 dB	4 dB	10 dB	16 dB	20 dB
L_1-pen.	*18.3 ± 8.8**	*4.6 ± 2.6**	*1.2 ± 0.7*$^{*\times}$	*0.5 ± 0.3**	*3.7 ± 1.4*$^{*\times}$	*1.0 ± 0.4*$^{*\times}$	*0.2 ± 0.1**	*1.0 ± 0.03**
GM	94.1 ± 108.3*	8.8 ± 13.1	3.1 ± 2.4*	2.7 ± 1.4*	12.5 ± 5.1	3.4 ± 1.2*	2.7 ± 0.5*	2.6 ± 0.3*
AKSM	**13.0 ± 5.6**	**3.6 ± 1.6**	**0.9 ± 0.5**	**0.3 ± 0.2**	**1.6 ± 0.6**	**0.4 ± 0.2**	**0.1 ± 0.04**	**0.04 ± 0.01**

parameter, and again, the cross-point between both curves represents a good working point for applications where both detection rates are similarly relevant. From this point of view, the detection performance of the algorithms is qualitatively similar to their estimation performance, that is, the highest for AKSM (close to 100% for Se and Sp simultaneously), and for the L_1-penalized and the GM methods. The value of the free parameter for optimal detection was again close, but not identical, to the value of the free parameter for optimal estimation in all the algorithms.

Records of estimated signals show a high variability, but in Figure 3, we present some representative results of those corresponding to previously presented performance figures. Optimum values of the free parameters in this example were used. It can be observed that L_1-penalized deconvolution detects almost all the peaks, but it produces a number of low amplitude spurious peaks. In the example, the GM procedure reaches a high estimation performance and a low number (yet high amplitude) of spurious peaks, as well as a misdetection of a large peak. Finally, the main drawback that could be observed for AKSM is the systematic error in the peak amplitude detection, due to a collateral effect of insensitivity caused by using $\varepsilon \neq 0$ for yielding sparseness on the ε-Huber cost function, but total detection was often attained with almost no spurious detection.

With respect to sparseness of the tested algorithms, it can be analyzed from the merit figures for estimation and for detection. The values of F^{null} can be seen as an averaged measurement of sparseness of each SD method, taking into account that similarly high values of this parameter can be given either by a large number of spurious peaks with low amplitude or by a lower number of spurious peaks with high amplitude. Note that, in any case, the AKSM algorithm reaches much better values, specially when compared to GM. Similar conclusions with respect to sparseness were obtained from the sensitivity and specificity results.

5.5. Robustness with respect to non-Gaussian noise

Table 2 shows the estimation performance of the compared algorithms for Laplacian and uniform noise sources added to the deterministic signal instead of a Gaussian noise. Optimal-free parameters were previously estimated for all the algorithms. Again, Wilcoxon-paired test was used ($*$ denotes $p < .01$ for AKSM versus GM and for AKSM versus $L1$ comparisons, $\times$ denotes $p < .01$ for GM versus $L1$ comparison). In general, estimation capabilities of each algorithm are similar for Gaussian and Laplacian noises, and dramatically higher for uniform noise. The L_1-penalized and the AKSM algorithms still exhibits good performance in the presence of non-Gaussian noise, whereas the GM algorithm shows slightly poorer performance, specially for low SNR. It can be seen that in the presence of both Laplacian and uniform noise, MSE is lower for ASKM in all cases.

5.6. An application example with real data

We analyze a B-scan given by an ultrasonic transducer array from a layered composite material. An A-scan is the ultrasound echo signal received by a single element in the transducer array, and the set of A-scans obtained by each of the elements of the array is known as B-scan. Hence each A-scan is a time-varying signal, whereas B-scan is a spatiotemporal representation of the ultrasound. Scan data are available at http://www.signal.uu.se/Toolbox/DT/ and

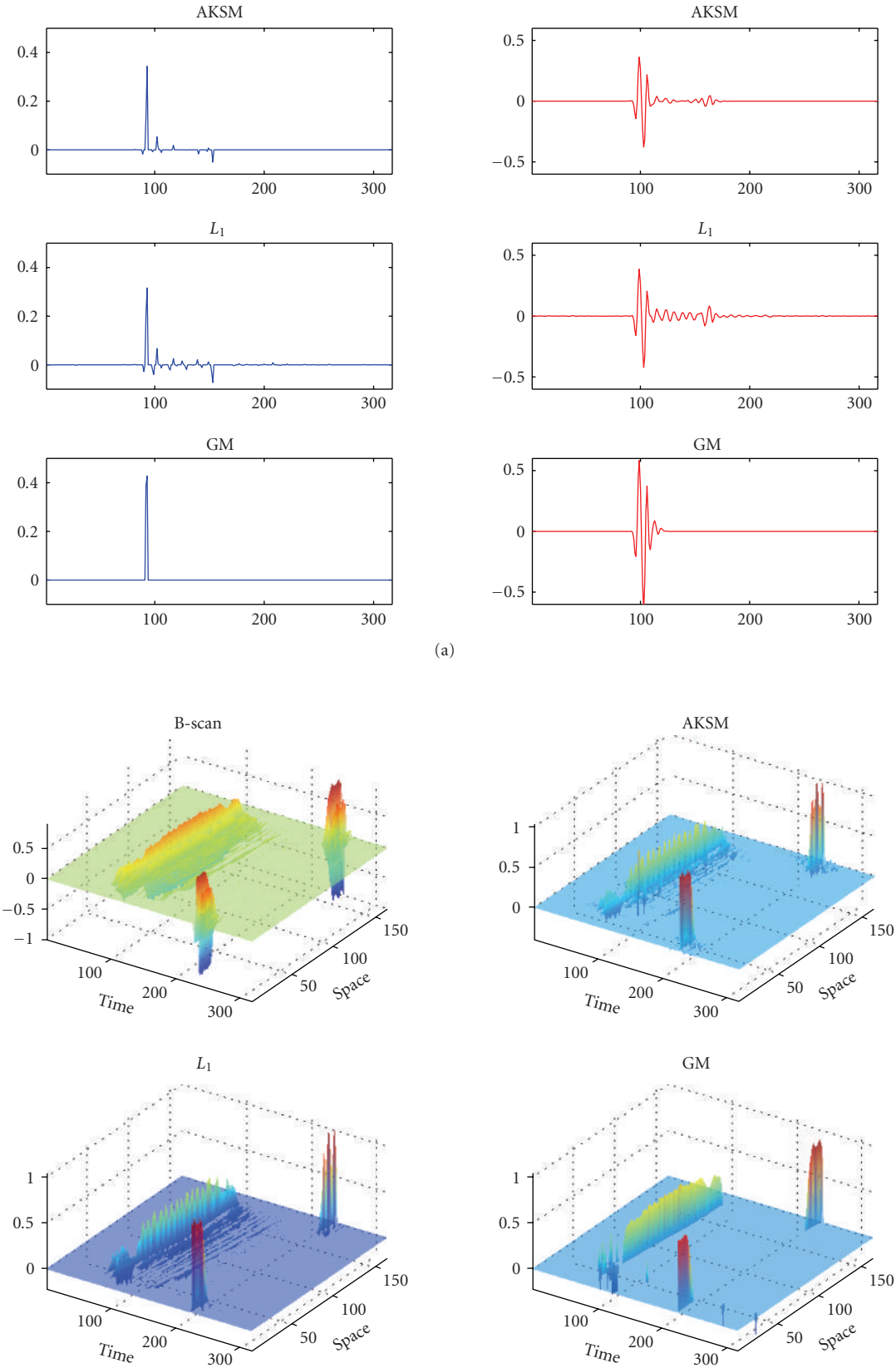

FIGURE 4: Real data application: deconvolution of the impulse response of a ultrasound transducer in the analysis of a layered composite material. (a) Example of deconvolution of a single A-scan line with each algorithm: left, sparse estimated signal; right, estimated A-line. (b) Deconvolution of the B-scan data.

details on this application can be found elsewhere [42]. Figure 4(a) shows a signal example (A-scan) of the sparse signal estimated by each of the tested methods, using as impulse response a prototype obtained from the B-scan in a single line where clearly only one reflection was present. The same panel also shows the reconstructed observed signal. Noise level in the data was relatively low, and free parameters of each algorithm were adjusted accordingly to the results obtained with the synthetic signal example (for this purpose, a normalization of the B-scan is previously made). It can be seen a behavior qualitatively similar to the previously seen for each algorithm, that is, the L_1 algorithm yields a good-quality solution with a noticeably number of spurious peaks, the AKSM algorithm yields a good-quality solution with less spurious peaks, and the GM algorithm fails at detecting the low amplitude peaks that are clearly present in this A-scan signal. Figure 4(b) shows the reconstruction of the complete B-scan data, with similar behavior for all the A-scan lines.

6. CONCLUSIONS

A fully practical algorithm for SD using SVM principles, which we call the AKSM algorithm, has been introduced and evaluated. It works with the convolution of the observed sequence with the time-reversed impulse response, this way creating an implicit autocorrelation kernel problem. The use of an ε-Huber cost function in the proposed algorithm provides a high degree of robustness against the presence of non-Gaussian additive noises. The solution is sparse as a consequence of the support vector formulation, and the implicit regularization of all SVM algorithms helps us to obtain interesting results. These advantages can be observed in illustrative simulation setups, which serves to compare the new algorithm performance with those of two well-known procedures, the L_1-penalized and the GM model deconvolution algorithms. The proposed algorithm presents an advantage in terms of both estimation and detection capabilities in most the situations, and there is no difficulty in finding appropriate values for its hyperparameters.

Further effort will be devoted to expand the SVM principles presented here to obtain improved algorithms in other relevant signal and image-processing problems, specially those useful for spectral analysis and image denoizing and deblurring.

APPENDICES

A. PSM DUAL PROBLEM

Instead of making the full derivation of the PSM algorithm in (15), a fast method can be used from the linear signal-processing framework in [28]. In brief, be $\{y[n]\}$ a discrete time series, observations with noise, and be $\{z_p[n],\ p = 0,\ldots,N\}$ a vector basis of the reconstruction signal space. Then the general linear model can be expressed as

$$y[n] = \sum_{p=0}^{N} w_p z_p[n] + e[n]. \tag{A.1}$$

The linear framework for SVM linear signal models requires to determine, at the begining, the transversal vector of the signal space basis, given by $\mathbf{v}_s = [z_0[s], z_1[s], \ldots, z_N[s]]^T$; then, to obtain the elements of the SV algorithm is straightforward. In our case, we can see from (1) that the signal space is generated by the time-shifted versions of the impulse response, that is, $z_p[n] = h[n-p]$, and then, the pth coefficient vector of the linear model is just the pth sample of the unknown sparse signal, that is, $w_p = \hat{x}[p]$. Therefore, the transversal vector is just $\mathbf{v}_s = [h[s], h[s-1], h[s-2], \ldots, h[s-N]]^T$, so that the dot product matrix is given by $R(i,l) = \mathbf{v}_i^T, \mathbf{v}_l = \sum_{k=0}^{P} h[i-k]h[l-k]$ for this problem statement, and the unknown signal is given by (16).

Then the dual problem consists on maximizing

$$-\frac{1}{2}(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)^T(\mathbf{R}+\gamma\mathbf{I})(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*) + (\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)^T\mathbf{y} - \varepsilon\mathbf{1}^T(\boldsymbol{\alpha}+\boldsymbol{\alpha}^*) \tag{A.2}$$

constrained to $0 \le \boldsymbol{\alpha}^{(*)} \le C$, where $\mathbf{1}$ denotes an all ones column vector, $\mathbf{I}$ denotes the identity matrix, $\alpha[n]$, $\alpha[n]^*$ are the Lagrange multipliers associated to the positive and negative parts of residuals $e[n]$ as usual in SVM, $\boldsymbol{\alpha}^{(*)} = [\alpha_0^{(*)}, \ldots, \alpha_N^{(*)}]^T$, and $\mathbf{y} = [y[0], \ldots, y[N]]^T$. This is a quadratic programming (QP) problem, from which solution (16) is obtained.

B. AKSM DUAL PROBLEM

As usual in SVM regression algorithms, the optimization of (23) leads to a dual problem to be optimized, which yields model coefficients $\eta[n]$ from Lagrange multipliers $\alpha[n]$, $\alpha[n]^*$, that are associated to residuals $e[n]$. In this case, the elements of the Gram matrix of the problem are found to be

$$T(i,l) = \boldsymbol{\phi}(i)^T\boldsymbol{\phi}(l) = K(i,l) = R^h[i-l], \tag{B.1}$$

and accordingly, the dual problem can be easily shown to consist on the maximization of

$$-\frac{1}{2}(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)^T(\mathbf{T}+\gamma\mathbf{I})(\boldsymbol{\alpha}-\boldsymbol{\alpha}^*) + (\boldsymbol{\alpha}-\boldsymbol{\alpha}^*)^T\mathbf{z} - \varepsilon\mathbf{1}^T(\boldsymbol{\alpha}+\boldsymbol{\alpha}^*) \tag{B.2}$$

subject to $0 \le \boldsymbol{\alpha}^{(*)} \le C$, which is formally identical to the QP problem in (A.2). A detailed derivation of the dual problem can be seen in [31] for the signal-processing application of discrete time series interpolation.

ACKNOWLEDGMENT

This work has been partly supported by research projects S-0505/TIC/0223 (Madrid Government), TEC2007-68096-C02/TCM, and TEC2005-00992, ESP2005-07724-C05-03, TEC2006-13845/TCM, and CONSOLIDER/CSD2007-00018 (Spanish MEC).

REFERENCES

[1] J. Mendel and C. S. Burrus, *Maximum-Likelihood Deconvolution: A Journey into Model-Based Signal Processing*, Springer, New York, NY, USA, 1990.

[2] M. S. O'Brien, A. N. Sinclair, and S. M. Kramer, "Recovery of a sparse spike time series by L_1 norm deconvolution," *IEEE Transactions on Signal Processing*, vol. 42, no. 12, pp. 3353–3365, 1994.

[3] N. P. Galatsanos, A. K. Katsaggelos, R. T. Chin, and A. D. Hillery, "Least squares restoration of multichannel images," *IEEE Transactions on Signal Processing*, vol. 39, no. 10, pp. 2222–2236, 1991.

[4] W. S. Ellis, S. J. Eisenberg, D. M. Auslander, M. W. Dae, A. Zakhor, and M. D. Lesh, "Deconvolution: a novel signal processing approach for determining activation time from fractionated electrograms and detecting infarcted tissue," *Circulation*, vol. 94, no. 10, pp. 2633–2640, 1996.

[5] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*, Prentice Hall, Englewood Cliffs, NJ, USA, 1993.

[6] B. Rice, "Inverse convolution filters," *Geophysics*, vol. 27, no. 1, pp. 4–18, 1962.

[7] J. Claerbout and E. A. Robinson, "The error in least-squares inverse filtering," *Geophysics*, vol. 29, no. 1, pp. 118–120, 1964.

[8] W. T. Ford and J. H. Hearne, "Least-squares inverse filtering," *Geophysics*, vol. 31, no. 5, pp. 917–926, 1966.

[9] A. J. Berkhout, "Least-squares inverse filtering and wavelet deconvolution," *Geophysics*, vol. 42, no. 7, pp. 1369–1383, 1977.

[10] A. Tikhonov and V. Y. Arsenin, *Solutions to Ill-Posed Problems*, Winston & Sons, Washington, DC, USA, 1977.

[11] A. E. Hoerl and R. W. Kennard, "Ridge regression: biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55–67, 1970.

[12] S. Huffel and J. Vanderwalle, *The Total Least Squares Problem: Computational Aspects and Analysis*, SIAM, Philadelphia, Pa, USA, 1991.

[13] Y. C. Eldar and A. V. Oppenheim, "Covariance shaping least-squares estimation," *IEEE Transactions on Signal Processing*, vol. 51, no. 3, pp. 686–697, 2003.

[14] C. Narduzzi, "Inverse filtering with signal-adaptive constraints," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 4, pp. 1553–1559, 2005.

[15] H. Taylor, S. Banks, and J. F. McCoy, "Deconvolution with the L_1 norm," *Geophysics*, vol. 44, no. 1, pp. 39–52, 1979.

[16] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Review*, vol. 43, no. 1, pp. 129–159, 2001.

[17] J. Kormylo and J. Mendel, "Maximum likelihood detection and estimation of Bernoulli-Gaussian processes," *IEEE Transactions on Information Theory*, vol. 28, no. 3, pp. 482–488, 1982.

[18] I. Santamaría-Caballero, C. J. Pantaleón-Prieto, and A. Artés-Rodríguez, "Sparse deconvolution using adaptive mixed-Gaussian models," *Signal Processing*, vol. 54, no. 2, pp. 161–172, 1996.

[19] Y. C. Eldar and N. Merhav, "A competitive minimax approach to robust estimation of random parameters," *IEEE Transactions on Signal Processing*, vol. 52, no. 7, pp. 1931–1946, 2004.

[20] Y. C. Eldar, A. Ben-Tal, and A. Nemirovski, "Linear minimax regret estimation of deterministic parameters with bounded data uncertainties," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2177–2188, 2004.

[21] Y. C. Eldar, A. Ben-Tal, and A. Nemirovski, "Robust mean-squared error estimation in the presence of model uncertainties," *IEEE Transactions on Signal Processing*, vol. 53, no. 1, pp. 168–181, 2005.

[22] Y. C. Eldar, "Robust deconvolution of deterministic and random signals," *IEEE Transactions on Information Theory*, vol. 51, no. 8, pp. 2921–2929, 2005.

[23] R. Neelamani, H. Choi, and R. Baraniuk, "ForWaRD: fourier-wavelet regularized deconvolution for ill-conditioned systems," *IEEE Transactions on Signal Processing*, vol. 52, no. 2, pp. 418–433, 2004.

[24] S. A. Kassam and H. V. Poor, "Robust techniques for signal processing: a survey," *Proceedings of the IEEE*, vol. 73, no. 3, pp. 433–481, 1985.

[25] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, New York, NY, USA, 1998.

[26] G. Camps-Valls, J. L. Rojo-Álvarez, and M. Martínez-Ramón, Eds., *Kernel Methods in Bioengineering, Signal, and Image Processing*, Idea Group, Hershey, Pa, USA, 2007.

[27] J. Mercer, "Functions of negative and positive type and their connection with the theory of integral equations," *Philosophical Transactions of the Royal Society of London A*, vol. 209, pp. 415–446, 1909.

[28] J. L. Rojo-Álvarez, G. Camps-Valls, M. Martínez-Ramón, E. Soria-Olivas, A. Navia-Vázquez, and A. R. Figueiras-Vidal, "Support vector machines framework for linear signal processing," *Signal Processing*, vol. 85, no. 12, pp. 2316–2326, 2005.

[29] J. L. Rojo-Álvarez, M. Martínez-Ramón, M. de Prado-Cumplido, A. Artés-Rodríguez, and A. R. Figueiras-Vidal, "Support vector method for robust ARMA system identification," *IEEE Transactions on Signal Processing*, vol. 52, no. 1, pp. 155–164, 2004.

[30] G. Camps-Valls, L. Bruzzone, J. L. Rojo-Álvarez, and F. Melgani, "Robust support vector regression for biophysical variable estimation from remotely sensed images," *IEEE Geoscience and Remote Sensing Letters*, vol. 3, no. 3, pp. 339–343, 2006.

[31] J. L. Rojo-Álvarez, C. Figuera-Pozuelo, C. E. Martínez-Cruz, G. Camps-Valls, F. Alonso-Atienza, and M. Martínez-Ramón, "Nonuniform interpolation of noisy signals using support vector machines," *IEEE Transactions on Signal Processing*, vol. 55, no. 8, pp. 4116–4126, 2007.

[32] F. Girosi, M. Jones, and T. Poggio, "Regularization theory and neural networks architectures," *Neural Computation*, vol. 7, no. 2, pp. 291–269, 1995.

[33] T. Poggio and S. Smale, "The mathematics of learning: dealing with data," *Notices of the American Mathematical Society*, vol. 50, no. 5, pp. 537–544, 2003.

[34] M. A. Aizerman, E. M. Braverman, and L. Rozoner, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and Remote Control*, vol. 25, pp. 821–837, 1964.

[35] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, Cambridge, UK, 2004.

[36] L. Zhang, W. Zhou, and L. Jiao, "Wavelet support vector machine," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 34, no. 1, pp. 34–39, 2004.

[37] J. Suykens, "Support vector machines: a nonlinear modelling and control perspective," *European Journal of Control*, vol. 7, no. 2-3, pp. 311–327, 2001.

[38] D. Mattera and S. Haykin, "Support vector machines for dynamic recostruction of chaotic systems," in *Advances in Kernel Methods*, pp. 211–242, MIT Press, Cambridge, Mass, USA, 1999.

[39] A. R. Figueiras-Vidal, D. Docampo-Amoedo, J. R. Casar-Corredera, and A. Artés-Rodríguez, "Adaptive iterative algorithms for spiky deconvolution," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, no. 8, pp. 1462–1466, 1990.

[40] J. T. Kwok and I. W. Tsang, "Linear dependency between ε and the input noise in ε-support vector regression," *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 544–553, 2003.

[41] V. Cherkassky and Y. Ma, "Practical selection of SVM parameters and noise estimation for SVM regression," *Neural Networks*, vol. 17, no. 1, pp. 113–126, 2004.

[42] T. Olofsson, "Semi-sparse deconvolution robust to uncertainties in the impulse responses," *Ultrasonics*, vol. 42, no. 1–9, pp. 969–975, 2004.

EURASIP Journal on Advances in Signal Processing

Special Issue on Microphone Array Speech Processing

Call for Papers

Significant knowledge about microphone arrays has been gained from years of intense research and product development. There have been numerous applications suggested, for example, from large arrays (on the order of >100 elements) for use in auditoriums to small arrays with only 2 or 3 elements for hearing aids and mobile telephones. Apart from that, array technology has been widely applied in the areas of speech recognition and more recently surveillance. Traditional techniques that have been used for microphone arrays include the fixed spatial filter as well as optimal and adaptive beamforming. These techniques model input or calibration signals as well as localization information for their design. Today contemporary techniques using blind signal separation (BSS) and time frequency masking techniques have attracted significant attraction. Those techniques are less reliant on array modeling and localization, but more on the statistical properties of speech signals such as sparseness, non-Gaussianity, nonstationarity, and so forth. The main advantage that multiple microphones add from a theoretical perspective is the spatial diversity, which is an effective tool to combat interference, reverberation, and noise when used according to the theoretical assumption. Combining spatial information with time-frequency information and perceptual cues will lead to innovative techniques and new methods, which will provide improved communication capabilities in challenging acoustic environments.

To further enhance current research and to promote new applications, this special issue aims to collect and present the latest research efforts in signal processing methods and algorithms for microphone arrays.

Topics of interest include, but are not limited to:

- Optimal and adaptive beamforming
- Blind signal extraction methods
- Multichannel dereverberation techniques
- Microphone array-assisted multichannel acoustic echo cancellation
- Spatial filtering techniques
- Sound source localization and tracking
- Psychoacoustically motivated procedures and algorithms such as perceptual cues, hearing thresholds, and spatial masking effects
- Distributed microphone networks
- Spherical array of microphones and Eigen/Modal beamforming

Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://www.hindawi.com/journals/asp/guidelines.html. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/, according to the following timetable:

Manuscript Due	August 1, 2009
First Round of Reviews	November 1, 2009
Publication Date	February 1, 2010

Guest Editors

Sven Nordholm, Western Australian Telecommunications Research Institute (WATRI), Curtin University of Technology, Perth, WA 6009, Australia; sven@watri.org.au

Thushara Abhayapala, Department of Information Engineering, Research School of Information Sciences and Engineering, ANU College of Engineering and Computer Sciences, The Australian National University, Canberra, ACT 0200, Australia; thushara.abhayapala@anu.edu.au

Simon Doclo, NXP Semiconductors, Corporate I&T-Research, Interleuvenlaan 80, 3001 Leuven, Belgium; simon.doclo@nxp.com

Sharon Gannot, School of Engineering, Bar-Ilan University, 52900 Ramat-Gan, Israel; gannot@macs.biu.ac.il

Patrick Naylor, Department of Electrical and Electronic Engineering, Imperial College, London SW7 2AZ, UK; p.naylor@imperial.ac.uk

Ivan Tashev, Microsoft Research Redmond, One Microsoft Way, Redmond WA, 98052-6399, USA; ivantash@microsoft.com

EURASIP Journal on Image and Video Processing

Special Issue on

Multicamera Information Processing: Acquisition, Collaboration, Interpretation, and Production

Call for Papers

Video sensors have gained in resolution, quality, cost-efficiency, and ease of use during the last decade, thus fostering the deployment of rich acquisition settings, to cheaply and effectively capture scenes at high spatiotemporal resolution, in multiple locations and directions. By providing extended and redundant coverage, multicamera imaging provides a practical approach to support robust scene interpretation, integrated situation awareness, as well as rich interactive and immersive experience in many different areas of industry, health-care, education, and entertainment. Tools and algorithms that aim to recognize high-level semantic concepts and their spatiotemporal and causal relations directly depend on the robustness and reliability of the underlying detection and tracking methods. These tasks related to scene interpretation have a strong impact on many real-life applications and are also fundamental to understand how to render a scene, for example, in a sport event summarization context or while browsing multiview video surveillance content. Finally, multiview imaging allows for immersive visualization by adapting rendered images to display capabilities and/or viewer requests. The goal of this special issue is to present the recent theoretical and practical advances that take advantage of multiview processing to improve 3D scene monitoring, immersive rendering, and (semi-)automatic content creation. Topics of interest include, but are not limited to:

- Acquisition of multiview and 3D images
- Multicamera information fusion
- Automated extraction of calibration or geometry information
- Distributed scene representation and communication
- Depth estimates and arbitrary view synthesis
- Multiview object detection and tracking
- Multiview video stream events/activities mining
- Multiview event detection and recognition
- Assistance to interactive video browsing in a distributed surveillance camera network
- Immersive rendering, and 3D scene virtual navigation
- Automatic and/or personalized summarization of sports events
- Plants or impaired people monitoring applications
- Advanced application case studies

Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://www.hindawi.com/journals/ivp/guidelines.html. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/ according to the following timetable:

Manuscript Due	December 1, 2009
First Round of Reviews	March 1, 2010
Publication Date	June 1, 2010

Lead Guest Editor

Christophe De Vleeschouwer, UCL, Louvain-la-Neuve, Belgium; christophe.devleeschouwer@uclouvain.be

Guest Editors

Andrea Cavallaro, Queen Mary, University of London, London, UK; andrea.cavallaro@elec.qmul.ac.uk

Pascal Frossard, EPFL, Lausanne, Switzerland; pascal.frossard@epfl.ch

Li-Qun Xu, British Telecommunications PLC, London, UK; li-qun.xu@bt.com

Peter Tu, GE Global Research, Niskayuna, NY, USA; tu@crd.ge.com

International Journal of Digital Multimedia Broadcasting

Special Issue on
Video Analysis, Abstraction, and Retrieval: Techniques and Applications

Call for Papers

The proliferation of TV broadcast channels and programs has led to an explosion of digital video content, which results in large personal and public video databases. However, the rapidly increasing availability of video data has not yet been accompanied by an increase in its accessibility. This is due to the situation that video data are naturally different to traditional forms of data, which can be easily accessed and searched based on text. Therefore, how to efficiently organize broadcast video, such as TV news and sports, into more compact forms and extract semantically meaningful information becomes more and more important. In the past ten years, the majority of research has gradually converged to three fundamental areas, namely, video analysis, video abstraction, and video retrieval. Video analysis is utilized to extract both general and domain-specific visual features, such as color, texture, shape, human faces, and human motion. Video abstraction is to generate a representation of visual information, which is similar to the extraction of keywords or summaries in text document processing. Basically, video abstraction is associated with key-frame detection, shot clustering, and the extraction of domain knowledge of the targeted video source. The content attributes found in video analysis and abstraction processes are often referred to as metadata. In many information systems, we need fast schemes and tools to use content metadata to query, search, and browse large video databases. Although a lot of efforts have been devoted into this area, both computational cost and accuracy of the existing systems are still far from satisfactory.

This special issue aims at capturing the latest advances of the research community working in video analysis, abstraction, and retrieval for broadcasting applications. The objectives of this special issue are twofold: (1) publishing novel fundamental techniques, and (2) showcasing robust systems to treat popular broadcast videos, such as TV news and sports video. Topics of interest include, but are not limited to:

- Feature extraction and description from broadcast video
- Object detection, tracking, and recognition in broadcast video
- Shot boundary detection and scene segmentation
- Key frame extraction and video summarization
- Efficient methods for video indexing and concepts modeling
- Semantic content understanding and recognition
- Video browsing/visualization tools for the broadcast video
- Semantic annotations of video content
- Metadata Standards for Video Analysis, Abstraction and Retrieval
- Multimodal data generation and fusion
- User interface for media browsing and search
- General framework for video retrieval
- Evaluation techniques and methodologies for video abstraction and retrieval
- Robust systems: TV news, sports, and so forth.

Before submission authors should carefully read over the journal's Author Guidelines, which are located at http://www.hindawi.com/journals/ijdmb/guidelines.html. Prospective authors should submit an electronic copy of their complete manuscript through the journal Manuscript Tracking System at http://mts.hindawi.com/, according to the following timetable:

Manuscript Due	September 1, 2009
First Round of Reviews	December 1, 2009
Publication Date	March 1, 2010

Guest Editors

Jungong Han, Eindhoven University of Technology, 5600 MB Eindhoven, The Netherlands; jg.han@tue.nl

Ling Shao, Philips Research Laboratories, 5656 AA Eindhoven, The Netherlands; l.shao@philips.com

Peter H. N. de With, CycloMedia/Eindhoven University of Technology, 4180 BB Waardenburg, The Netherlands; p.h.n.de.with@tue.nl

Ling Guan, Ryerson University, Toronto, ON, Canada M5B 2K3; lguan@ee.ryerson.ca