6TH IMA CONFERENCE ON MATHEMATICS IN SIGNAL PROCESSING, CIRENCESTER, UK, DECEMBER 2004

A MULTISCALE SMOOTHING ALGORITHM FOR PAIRWISE MARKOV TREES

Jean Lecomte, François Desbouvries and Wojciech Pieczynski

GET / INT / dépt. CITI and CNRS UMR 5157
9 rue Charles Fourier, 91011 Evry, France
Francois.Desbouvries@int-evry.fr

ABSTRACT

An important problem in multiresolution analysis of signals and images consists in restoring continuous hidden random variables $\mathbf{x} = \{\mathbf{x}_s\}_{s\in\mathcal{S}}$ from observed ones $\mathbf{y} = \{\mathbf{y}_s\}_{s\in\mathcal{S}}$. This is done classically in the context of Hidden Markov Trees (HMT). HMT have been generalized recently to Pairwise Markov Trees (PMT). In this paper we propose a smoothing restoration algorithm for Gaussian PMT.

1. INTRODUCTION

An important problem in signal and image processing consists in estimating hidden random variables $\mathbf{x} = \{\mathbf{x}_s\}_{s\in\mathcal{S}}$ from observed ones $\mathbf{y} = \{\mathbf{y}_s\}_{s\in\mathcal{S}}$. To that end, Bayesian restoration algorithms have been developped, mainly in the framework of Hidden Markov Models (HMM).

On the other hand, it is well known that if $(\mathbf{x}, \mathbf{y})$ is a classical HMM, then the pair $(\mathbf{x}, \mathbf{y})$ itself is Markovian. Conversely, starting from the sole assumption that $(\mathbf{x}, \mathbf{y})$ is Markovian, i.e. that $(\mathbf{x}, \mathbf{y})$ is a so-called Pairwise Markov Model (PMM), is a more general point of view which nevertheless enables the development of similar restoration algorithms. More precisely, some of the classical Bayesian restoration algorithms used in Hidden Markov Fields (HMF), Hidden Markov Chains (HMC) or Hidden Markov Trees (HMT), have been generalized recently to the more general frameworks of Pairwise Markov Fields (PMF) [1], Pairwise Markov Chains (PMC) with discrete [2] [3] or continuous [4] [5] state process, and of Pairwise Markov Trees (PMT) with discrete [6] [7] or continuous [7] [8] hidden variables.

From now on, we shall focus more specifically on multiresolution analysis and multiscale algorithms, which are of interest in a large variety of signal and image processing problems (see e.g. [9] [10] [11] [12] [13] [14] [15] as well as the tutorial [16]). Efficient restoration algorithms have been developed, under the assumption that the stochastic interactions of $\mathbf{x}$ and $\mathbf{y}$ are modeled by a hidden Markov tree with independent noise (HMT-IN) [9] [10] [11] [15].

In this paper we propose a smoothing Kalman-like restoration algorithm for Gaussian PMT. In a PMT the hidden tree $\mathbf{x}$ is not necessarily Markovian, and the observations $\mathbf{y}$ are not necessarily related to $\mathbf{x}$ as simply as in the HMT-IN case. Yet the conditional law of $\mathbf{x}$ given $\mathbf{y}$ remains Markovian, which in turn enables us to propose an efficient restoration algorithm.

This restoration algorithm (as well as existing restoration algorithms for discrete [6] [7] or continuous [8] PMT) should also be recast in the framework of artificial intelligence, and in particular of belief propagation - like inference algorithms. It is well known [17] that the inference problem in an arbitrary belief network (BN) is an NP-hard problem, and fast algorithms are known to exist only for particular subclasses of BN. Now, the PMT structure we deal with can be seen in two different ways : as a joint structure $\mathbf{z}$, it is a tree, and thus a particular polytree [1], but as an extended structure $(\mathbf{x}, \mathbf{y})$, its topology is no longer that of a polytree; and yet our restoration algorithm is linear in the number of nodes.

The rest of this paper is organized as follows. In section 2 we briefly recall the three embedded HMT-IN, HMT and PMT models. In section 3 we propose a general restoration algorithm, which is an adaptation to the continuous case of the restoration algorithm developped in [7] for the case where the hidden variables $\mathbf{x}$ are discrete. Finally in section 4 we consider the particular case of Gaussian processes, in which case the algorithm of section 3 reduces to a Kalman-like smoothing algorithm.

2. MARKOVIAN MODELS FOR TREES : HMT-IN $\subset$ HMT $\subset$ PMT

Let us consider a tree structure indexed on a set $\mathcal{S}$ of indices. Notations are as follows : $\mathcal{S}_1 = \{r\}$, $\mathcal{S}_2$, ..., $\mathcal{S}_{n+1}$ are the successive "generations" of the tree; each node s

[1] i.e. a directed acyclic graph in which, between any two nodes s and t, there is at most one path in the underlying undirected graph.

(apart from the root node r) has exactly one father s^-; the children of node s are denoted by s^+, and the set of all its descendants by s^{++} (see fig. 1).

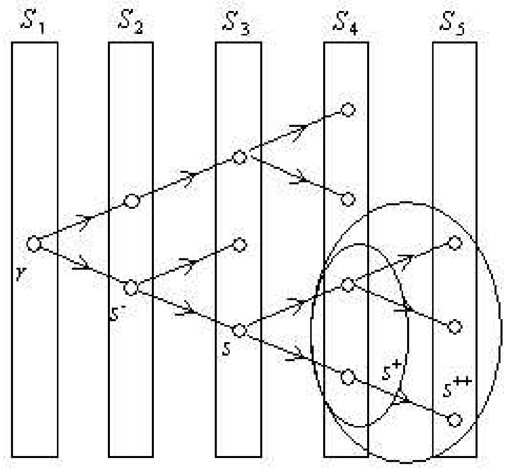

Fig. 1. The tree structure.

Let now $\mathbf{x} = \{\mathbf{x}_s\}_{s\in\mathcal{S}}$ and $\mathbf{y} = \{\mathbf{y}_s\}_{s\in\mathcal{S}}$ be two sets of random variables indexed on $\mathcal{S}$. Each $\mathbf{x}_s$ (resp. $\mathbf{y}_s$) belongs to $\mathbb{R}^p$ (resp. to $\mathbb{R}^q$). Let $p(\mathbf{x}_s)$ (resp. $p(\mathbf{y}_s)$) denote the probability density function (p.d.f.) of $\mathbf{x}_s$ (resp. of $\mathbf{y}_s$) w.r.t. Lebesgue measure, and for any set of indices Σ, let $p(\mathbf{x}_s|\{\mathbf{y}_\sigma\}_{\sigma\in\Sigma})$ denote the conditional p.d.f. of $\mathbf{x}_s$ given $\{\mathbf{y}_\sigma\}_{\sigma\in\Sigma}$. Other p.d.f. or conditional p.d.f. of interest are defined similarly.

The restoration algorithms developped in [9] [10] [11] [15] assume that the stochastic interactions between $\mathbf{x}$ and $\mathbf{y}$ are modeled by a classical HMT-IN model. In an HMT-IN model, $\mathbf{x}$ is a Markov Tree (MT), and conditionally on $\mathbf{x}$, the variables $\mathbf{y}_s$ are independent and satisfy $p(\mathbf{y}_s|\mathbf{x}) = p(\mathbf{y}_s|\mathbf{x}_s)$:

$$p(\mathbf{x},\mathbf{y}) = \underbrace{p(\mathbf{x}_r)\prod_{s\in\mathcal{S}\setminus\mathcal{S}_1} p(\mathbf{x}_s|\mathbf{x}_{s^-})}_{p(\mathbf{x})} \times \underbrace{\prod_{s\in\mathcal{S}} p(\mathbf{y}_s|\mathbf{x}_s)}_{p(\mathbf{y}|\mathbf{x})} . \quad (1)$$

Now, let us introduce the pair $\mathbf{z}_s = (\mathbf{x}_s, \mathbf{y}_s)$, and let $\mathbf{z} = \{\mathbf{z}_s\}_{s\in\mathcal{S}}$. A PMT model is a model in which we assume that $\mathbf{z}$ is an MT :

$$p(\mathbf{z}) = p(\mathbf{z}_r)\prod_{s\in\mathcal{S}\setminus\mathcal{S}_1} p(\mathbf{z}_s|\mathbf{z}_{s^-}). \quad (2)$$

Our interest for PMT comes from the following observation. A set of variables satisfying (1) also satisfies (2), so any HMT-IN is a PMT. However, the converse is not true, as can be seen at the local level : in a PMT the transition p.d.f. $p(\mathbf{z}_s|\mathbf{z}_{s^-})$ reads

$$\begin{aligned} p(\mathbf{z}_s|\mathbf{z}_{s^-}) &= p(\mathbf{x}_s,\mathbf{y}_s|\mathbf{x}_{s^-},\mathbf{y}_{s^-}) \\ &= p(\mathbf{x}_s|\mathbf{x}_{s^-},\mathbf{y}_{s^-})p(\mathbf{y}_s|\mathbf{x}_s,\mathbf{x}_{s^-},\mathbf{y}_{s^-}) \; ; \end{aligned}$$

so an HMT-IN is a particular PMT in which $p(\mathbf{x}_s|\mathbf{x}_{s^-},\mathbf{y}_{s^-})$ reduces to $p(\mathbf{x}_s|\mathbf{x}_{s^-})$ and $p(\mathbf{y}_s|\mathbf{x}_s,\mathbf{x}_{s^-},\mathbf{y}_{s^-})$ to $p(\mathbf{y}_s|\mathbf{x}_s)$. These simplications are rather rough, and making use of PMT enables to model more complex physical situations.

Let us also mention an intermediate model, which we call a Hidden Markov Tree (HMT), in which both $\mathbf{x}$ and $(\mathbf{x},\mathbf{y})$ are MT but the observations $\mathbf{y}_s$ are not necessarily independent conditionally on $\mathbf{x}$. Of course, any HMT-IN is an HMT, and any HMT is a PMT. However, PMT are more general than HMT, because if (2) holds, one can show that $\mathbf{x}$ is not necessarily a MT, as we see from the following result:

Proposition 1 *Let $\mathbf{z}$ be a dyadic PMT, i.e. a PMT in which each node s^- (which is not in the last generation $n+1$) has exactly two children s_1 and s_2. Assume that*

$$\textit{For all } s\in\mathcal{S}\setminus\mathcal{S}_1, \quad p(\mathbf{x}_s|\mathbf{x}_{s^-},\mathbf{y}_{s^-}) = p(\mathbf{x}_s|\mathbf{x}_{s^-}). \quad (3)$$

Then $\mathbf{x}$ is an MT. Conversely, assume that $\mathbf{x}$ is an MT, and that for all $s^- \in \mathcal{S}\setminus\mathcal{S}_{n+1}$, $p(\mathbf{z}_{s_1}|\mathbf{z}_s^-) = p(\mathbf{z}_{s_2}|\mathbf{z}_s^-)$, i.e. that conditionally on the father, the laws of the children are equal. Then (3) holds.

Proof of Proposition 1. A proof of Proposition 1 can be found in [6] (resp. in [8]) for the case where $\{\mathbf{x}_s\}_{s\in\mathcal{S}}$ are discrete (resp. continuous) random variables. ∎

3. A SMOOTHING ALGORITHM FOR CONTINUOUS PMT

From now on we shall assume that $\mathbf{z}$ is a PMT. We assume for simplicity that the tree is dyadic, and we denote by s_1 and s_2 the two children of s^-. The aim of this section consists in computing the posterior p.d.f. $p(\mathbf{x}_s|\mathbf{y})$ for an arbitrary $s \in \mathcal{S}$. Our smoothing-like algorithm is a two-step procedure : a first sweep (in the fine-to-coarse direction) computes $p(\mathbf{x}_r|\mathbf{y})$, and then a second sweep (in the coarse-to-fine direction) computes $p(\mathbf{x}_s|\mathbf{y})$ via a computational procedure which iterates along the path relating the root node r to node s.

Let us first show that $p(\mathbf{x}_s|\mathbf{x}_{s^-},\mathbf{y})$ can be computed recursively in the following way (the proof is omitted for want of space) :

Proposition 2 *Let*

$$\begin{aligned} \beta(\mathbf{x}_s) &= 1 \textit{ if } s\in\mathcal{S}_{n+1}, \\ &= p(\mathbf{y}_s^{++}|\mathbf{z}_s) \textit{ otherwise} . \end{aligned} \quad (4)$$

Then $\beta(\mathbf{x}_s)$ and $p(\mathbf{x}_s|\mathbf{x}_{s^-},\mathbf{y})$ can be computed recursively via the following (backward) recursions : for $m = n, n-$

$1, \cdots, 1$, *for all* $s^- \in \mathcal{S}_m$,

$$\beta(\mathbf{x}_{s^-}) = \prod_{i=1}^{2} \int p(\mathbf{z}_{s_i}|\mathbf{z}_{s^-})\beta(\mathbf{x}_{s_i})d\mathbf{x}_{s_i}, \quad (5)$$

$$p(\mathbf{x}_s|\mathbf{x}_{s^-}, \mathbf{y}) = p(\mathbf{x}_s|\mathbf{x}_{s^-}, \mathbf{y}_{s^-}, \mathbf{y}_s, \mathbf{y}_{s^{++}}) = \frac{p(\mathbf{z}_s|\mathbf{z}_{s^-})\beta(\mathbf{x}_s)}{\int p(\mathbf{z}_s|\mathbf{z}_{s^-})\beta(\mathbf{x}_s)d\mathbf{x}_s}. \quad (6)$$

We now address the computation of $p(\mathbf{x}_s|\mathbf{y})$ for a given node $s \in \mathcal{S}_m$. There is a unique path $\{\sigma_i\}_{i=1}^m$ (with $\sigma_1 = r$ and $\sigma_m = s$) relating node s to the root node r. Along this path, the conditional law of $\{\mathbf{x}_{\sigma_i}\}_{i=1}^m$ given $\mathbf{y}$ is Markovian. So $p(\mathbf{x}_s|\mathbf{y})$ can be computed as

$$p(\mathbf{x}_s|\mathbf{y}) = \int p(\mathbf{x}_r|\mathbf{y}) \prod_{i=2}^{m} p(\mathbf{x}_{\sigma_i}|\mathbf{x}_{\sigma_i^-}, \mathbf{y})d\mathbf{x}_{\sigma_1} \cdots d\mathbf{x}_{\sigma_{m-1}}, \quad (7)$$

in which $\{p(\mathbf{x}_{\sigma_i}|\mathbf{x}_{\sigma_i^-}, \mathbf{y}) =\}_{i=2}^m$ have been computed via (6), and

$$p(\mathbf{x}_r|\mathbf{y}) = \frac{p(\mathbf{z}_r)\beta(\mathbf{x}_r)}{\int p(\mathbf{z}_r)\beta(\mathbf{x}_r)d\mathbf{x}_r}. \quad (8)$$

4. A KALMAN-LIKE SMOOTHING ALGORITHM FOR GAUSSIAN PMT

The formulas of section 3, which hold irrespective of the law of $\mathbf{z}$, may prove difficult to compute in the general case. In the Gaussian case however, the computations can be carried out exactly and yield a Kalman-like smoothing algorithm, as we are going to see in this section [2].

Our assumptions are as follows. We assume that

$$\underbrace{\begin{bmatrix} \mathbf{x}_s \\ \mathbf{y}_s \end{bmatrix}}_{\mathbf{z}_s} = \underbrace{\begin{bmatrix} \mathbf{F}_s^1 & \mathbf{F}_s^2 \\ \mathbf{H}_s^1 & \mathbf{H}_s^2 \end{bmatrix}}_{\mathbf{F}_s} \begin{bmatrix} \mathbf{x}_{s^-} \\ \mathbf{y}_{s^-} \end{bmatrix} + \underbrace{\begin{bmatrix} \mathbf{G}_s^{11} & \mathbf{G}_s^{12} \\ \mathbf{G}_s^{21} & \mathbf{G}_s^{22} \end{bmatrix}}_{\mathbf{G}_s} \underbrace{\begin{bmatrix} \mathbf{u}_s \\ \mathbf{v}_s \end{bmatrix}}_{\mathbf{w}_s}, \quad (9)$$

in which $\mathbf{W} = \{\mathbf{w}_s\}_{s \in \mathcal{S} \setminus \mathcal{S}_1}$ are random vectors which are zero-mean, independent and independent of $\mathbf{z}_r$. Let us set

$$E(\mathbf{w}_s\mathbf{w}_s^T) = \mathbf{Q}_s, \widetilde{\mathbf{Q}}_s = \begin{bmatrix} \widetilde{\mathbf{Q}}_s^{11} & \widetilde{\mathbf{Q}}_s^{12} \\ \widetilde{\mathbf{Q}}_s^{21} & \widetilde{\mathbf{Q}}_s^{22} \end{bmatrix} = \mathbf{G}_s\mathbf{Q}_s\mathbf{G}_s^T. \quad (10)$$

We also assume that $\mathbf{W}$ is Gaussian and that $p(\mathbf{z}_r) \sim \mathcal{N}(\overline{\mathbf{z}}_r, \widetilde{\mathbf{Q}}_r)$. So $\mathbf{z}$ is a Gaussian process, and all conditionnal p.d.f. related to $\mathbf{z}$ are also Gaussian. As a consequence, the general recursions (5), (6) and (7) of section 3 reduce to recursions propagating the parameters (means and covariance matrices) of these Gaussian p.d.f., as we now see.

[2] Our algorithm could of course alternately be obtained as a recursive linear minimum mean square error restoration procedure; we chose to adopt the Gaussian point of view because the proofs are obtained in a simpler and more direct way.

We first begin with the following proposition, which gathers the results obtained by injecting the Gaussian assumption into Proposition 2 (the proof is omitted) :

Proposition 3 *Let us assume that* $\mathbf{z}$ *is a dyadic PMT and that model (9) holds. Suppose that* $p(\mathbf{z}_r) \sim \mathcal{N}(\overline{\mathbf{z}}_r, \widetilde{\mathbf{Q}}_r)$ *and that* $p(\mathbf{w}_s) \sim \mathcal{N}(\mathbf{0}, \mathbf{Q}_s)$.

Then for all $s^- \in \mathcal{S} \setminus \mathcal{S}_{n+1}$, *and for* $i = 1, 2$, *we have :*

$$\beta(\mathbf{x}_{s^-}) \sim \mathcal{N}(\underbrace{\begin{bmatrix} \widetilde{\mathbf{M}}_{s_1} \\ \widetilde{\mathbf{M}}_{s_2} \end{bmatrix}}_{\mathbf{M}_{s^-}} \mathbf{z}_{s^-}, \underbrace{\begin{bmatrix} \widetilde{\mathbf{C}}_{s_1}^{2,2} & \mathbf{0} \\ \mathbf{0} & \widetilde{\mathbf{C}}_{s_2}^{2,2} \end{bmatrix}}_{\mathbf{C}_{s^-}}), \quad (11)$$

$$p(\mathbf{x}_{s_i}|\mathbf{x}_{s^-}, \mathbf{y}) \sim \mathcal{N}(\mathbf{A}_{s_i|s^-}\mathbf{x}_{s^-} + \mathbf{b}_{s_i|s^-}, \mathbf{P}_{s_i|s^-}), \quad (12)$$

in which $\mathbf{M}_{s^-}$, $\mathbf{C}_{s^-}$, $\mathbf{A}_{s_i|s^-}$, $\mathbf{b}_{s_i|s^-}$ *and* $\mathbf{P}_{s_i|s^-}$ *can be computed sequentially through the following recursions :*

Initialization : for $s^- \in \mathcal{S}_n$ *and for* $i = 1, 2$,

$$\widetilde{\mathbf{M}}_{s_i} = [\mathbf{H}_{s_i}^1 \mathbf{H}_{s_i}^2], \quad (13)$$

$$\widetilde{\mathbf{C}}_{s_i}^{2,2} = \widetilde{\mathbf{Q}}_{s_i}^{2,2}, \quad (14)$$

$$\mathbf{A}_{s_i|s^-} = \mathbf{F}_{s_i}^1 - \widetilde{\mathbf{Q}}_{s_i}^{1,2}(\widetilde{\mathbf{Q}}_{s_i}^{2,2})^{-1}\mathbf{H}_{s_i}^1, \quad (15)$$

$$\mathbf{b}_{s_i|s^-} = \mathbf{F}_{s_i}^2\mathbf{y}_{s^-} + \widetilde{\mathbf{Q}}_{s_i}^{1,2}(\widetilde{\mathbf{Q}}_{s_i}^{2,2})^{-1}(\mathbf{y}_{s_i} - \mathbf{H}_{s_i}^2\mathbf{y}_{s^-}), \quad (16)$$

$$\mathbf{P}_{s_i|s^-} = \widetilde{\mathbf{Q}}_{s_i}^{1,1} - \widetilde{\mathbf{Q}}_{s_i}^{1,2}(\widetilde{\mathbf{Q}}_{s_i}^{2,2})^{-1}\widetilde{\mathbf{Q}}_{s_i}^{2,1}. \quad (17)$$

For $s^- \in \mathcal{S}_m$ *with* $1 \leq m \leq n-1$, *and for* $i = 1, 2$,

$$\widetilde{\mathbf{M}}_{s_i} = \begin{bmatrix} [\mathbf{0}\ \mathbf{I}]\mathbf{F}_{s_i} \\ \mathbf{M}_{s_i}\mathbf{F}_{s_i} \end{bmatrix}, \quad (18)$$

$$\widetilde{\mathbf{C}}_{s_i}^{2,2} = \begin{bmatrix} \widetilde{\mathbf{Q}}_{s_i}^{2,2} & [\widetilde{\mathbf{Q}}_{s_i}^{2,1}\widetilde{\mathbf{Q}}_{s_i}^{2,2}]\mathbf{M}_{s_i}^T \\ \mathbf{M}_{s_i}\begin{bmatrix} \widetilde{\mathbf{Q}}_{s_i}^{1,2} \\ \widetilde{\mathbf{Q}}_{s_i}^{2,2} \end{bmatrix} & \mathbf{C}_{s_i} + \mathbf{M}_{s_i}\widetilde{\mathbf{Q}}_{s_i}\mathbf{M}_{s_i}^T \end{bmatrix}, \quad (19)$$

$$\widetilde{\mathbf{C}}_{s_i}^{1,2} = \left[\widetilde{\mathbf{Q}}_{s_i}^{1,2}, [\widetilde{\mathbf{Q}}_{s_i}^{1,1}\widetilde{\mathbf{Q}}_{s_i}^{1,2}]\mathbf{M}_{s_i}^T\right], \quad (20)$$

$$\mathbf{A}_{s_i|s^-} = \mathbf{F}_{s_i}^1 - \widetilde{\mathbf{C}}_{s_i}^{1,2}(\widetilde{\mathbf{C}}_{s_i}^{2,2})^{-1}\begin{bmatrix} \mathbf{H}_{s_i}^1 \\ \mathbf{M}_{s_i}\begin{bmatrix} \mathbf{F}_{s_i}^1 \\ \mathbf{H}_{s_i}^1 \end{bmatrix} \end{bmatrix}, \quad (21)$$

$$\mathbf{b}_{s_i|s^-} = \mathbf{F}_{s_i}^2\mathbf{y}_{s^-} + \widetilde{\mathbf{C}}_{s_i}^{1,2}(\widetilde{\mathbf{C}}_{s_i}^{2,2})^{-1}(\begin{bmatrix} \mathbf{y}_{s_i} \\ \mathbf{y}_{s_i}^{++} \end{bmatrix} - \begin{bmatrix} \mathbf{H}_{s_i}^2 \\ \mathbf{M}_{s_i}\begin{bmatrix} \mathbf{F}_{s_i}^2 \\ \mathbf{H}_{s_i}^2 \end{bmatrix} \end{bmatrix}\mathbf{y}_{s^-}), \quad (22)$$

$$\mathbf{P}_{s_i|s^-} = \widetilde{\mathbf{Q}}_{s_i}^{1,1} - \widetilde{\mathbf{C}}_{s_i}^{1,2}(\widetilde{\mathbf{C}}_{s_i}^{2,2})^{-1}(\widetilde{\mathbf{C}}_{s_i}^{1,2})^T. \quad (23)$$

We now turn to the computation of $p(\mathbf{x}_{s_i}|\mathbf{y})$ (the proof is omitted) :

Proposition 4 *Let*

$$p(\mathbf{x}_{s_i}|\mathbf{y}) \sim \mathcal{N}(\mathbf{m}_{s_i}, \mathbf{P}_{s_i}). \quad (24)$$

Then $\mathbf{m}_{s_i}$ and $\mathbf{P}_{s_i}$ can be computed sequentially via the following recursions :

$$\widetilde{\mathbf{C}}_0^{1,2} = \left[\widetilde{\mathbf{Q}}_r^{1,2}, [\widetilde{\mathbf{Q}}_r^{1,1}\widetilde{\mathbf{Q}}_r^{1,2}]\mathbf{M}_r^T\right], \quad (25)$$

$$\widetilde{\mathbf{C}}_0^{2,2} = \begin{bmatrix} \widetilde{\mathbf{Q}}_r^{2,2} & [\widetilde{\mathbf{Q}}_r^{2,1}\widetilde{\mathbf{Q}}_r^{2,2}]\mathbf{M}_r^T \\ \mathbf{M}_r \begin{bmatrix} \widetilde{\mathbf{Q}}_r^{1,2} \\ \widetilde{\mathbf{Q}}_r^{2,2} \end{bmatrix} & \mathbf{C}_r + \mathbf{M}_r\widetilde{\mathbf{Q}}_r\mathbf{M}_r^T \end{bmatrix}, \quad (26)$$

$$\mathbf{m}_r = \overline{\mathbf{x}}_r + \widetilde{\mathbf{C}}_0^{1,2}(\widetilde{\mathbf{C}}_0^{2,2})^{-1}\left(\begin{bmatrix} \mathbf{y}_r \\ \mathbf{y}_r^{++} \end{bmatrix} - \begin{bmatrix} \overline{\mathbf{y}}_r \\ \mathbf{M}_r\overline{\mathbf{z}}_r \end{bmatrix}\right), \quad (27)$$

$$\mathbf{P}_r = \widetilde{\mathbf{Q}}_r^{1,1} - \widetilde{\mathbf{C}}_0^{1,2}(\widetilde{\mathbf{C}}_0^{2,2})^{-1}(\widetilde{\mathbf{C}}_0^{1,2})^T, \quad (28)$$

$$\mathbf{m}_{s_i} = \mathbf{A}_{s_i|s^-}\mathbf{m}_{s^-} + \mathbf{b}_{s_i|s^-}, \quad (29)$$

$$\mathbf{P}_{s_i} = \mathbf{P}_{s_i|s^-} + \mathbf{A}_{s_i|s^-}\mathbf{P}_{s^-}\mathbf{A}_{s_i|s^-}^T. \quad (30)$$

5. REFERENCES

[1] W. Pieczynski and A. N. Tebbache, "Pairwise Markov random fields and segmentation of textured images," *Machine Graphics and Vision*, vol. 9, no. 3, pp. 705–718, March 2000.

[2] W. Pieczynski, "Pairwise Markov chains," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 634–639, May 2003.

[3] S. Derrode and W. Pieczynski, "Signal and image segmentation using pairwise Markov chains," *IEEE Transactions on Signal Processing*, vol. 52, no. 9, pp. 2477–89, 2004.

[4] W. Pieczynski and F. Desbouvries, "Kalman filtering using pairwise Gaussian models," in *Proceedings of the Icassp*, Hong-Kong, April 6-10 2003, vol. 6, pp. VI–57 – VI–60.

[5] F. Desbouvries and W. Pieczynski, "Particle filtering in pairwise and triplet Markov chains," in *Proceedings of the IEEE - EURASIP Workshop on Nonlinear Signal and Image Processing (NSIP 2003)*, Grado-Gorizia, Italy, June 8-11 2003.

[6] W. Pieczynski, "Arbres de Markov couple - pairwise Markov trees," *Comptes Rendus de l'Académie des Sciences - Mathématiques*, vol. 335, pp. 79–82, 2002, Ser. I (*in French*).

[7] E. Monfrini, J. Lecomte, F. Desbouvries, and W. Pieczynski, "Image and signal restoration using pairwise Markov trees," in *Proceedings of the 2003 IEEE Workshop on Statistical Signal Processing*, St. Louis, MI, September 2003.

[8] F. Desbouvries and J. Lecomte, "Multiscale Bayesian estimation in pairwise Markov trees," in *Proceedings of the EUropean SIgnal Processing COnference (EUSIPCO)*, Vienna, Austria, September 2004.

[9] M. Basseville, A. Benveniste, K. C. Chou, S. A. Golden, R. Nikoukhah, and A. S. Willsky, "Modeling and estimation of multiresolution stochastic processes," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 766–84, March 1992.

[10] C. A. Bouman and M. Shapiro, "A multiscale random field model for Bayesian image segmentation," *IEEE Transactions on Image Processing*, vol. 3, no. 3, pp. 162–77, March 1994.

[11] K. C. Chou, A. S. Willsky, and A. Benveniste, "Multiscale recursive estimation, data fusion, and regularization," *IEEE Transactions on Automatic Control*, vol. 39, no. 3, pp. 464–78, March 1994.

[12] M. S. Crouse, R. D. Nowak, and R. G. Baraniuk, "Wavelet-based statistical signal processing using hidden Markov models," *IEEE Transactions on Signal Processing*, vol. 46, no. 4, pp. 886–902, April 1998.

[13] R. H. Riedi, M. S. Crouse, V. J. Ribeiro, and R. G. Baraniuk, "A multifractal wavelet model with application to network traffic," *IEEE Transactions on Information Theory*, vol. 45, no. 3, pp. 992–1018, April 1999.

[14] R. Nowak, "Multiscale hidden Markov models for Bayesian image analysis," in *Bayesian inference in wavelet based models*, B. Vidakovic and P. Müller, Eds. Springer-Verlag, New-York, 1999.

[15] J. M. Laferté, P. Perez, and F. Heitz, "Discrete Markov image modeling and inference on the quadtree," *IEEE Transactions on Image Processing*, vol. 9, no. 3, pp. 390–404, March 2000.

[16] A. S. Willsky, "Multiresolution Markov models for signal and image processing," *Proceedings of the IEEE*, vol. 90, no. 8, pp. 1396–1458, August 2002.

[17] G. F. Cooper, "The computational complexity of probabilistic inference using Bayesian belief networks," *Artificial Intelligence*, vol. 42, pp. 393–405, 1990.