Modeling Trajectory of Dynamic Clusters in Image Time-Series for Spatio-Temporal Reasoning

Patrick Héas and Mihai Datcu, *Senior Member, IEEE*

***Abstract*—During the last decades, satellites have acquired incessantly high-resolution images of many Earth observation sites. New products have arisen from this intensive acquisition process: high-resolution satellite image time-series (SITS). They represent a large data volume with a rich information content and may open a broad range of new applications. This paper presents an information mining concept which enables a user to learn and retrieve spatio-temporal structures in SITS. The concept is based on a hierarchical Bayesian modeling of SITS information content which enables us to link the interest of a user to specific spatio-temporal structures. The hierarchy is composed of two inference steps: an unsupervised modeling of dynamic clusters resulting in a graph of trajectories, and an interactive learning procedure based on graphs which leads to the semantic labeling of spatio-temporal structures. Experiments performed on a SPOT image time-series demonstrate the concept capabilities.**

***Index Terms*—Bayesian modeling, dynamic cluster trajectories, information mining, semantic labeling, spatio-temporal learning.**

I. Introduction

A. Satellite Image Time-Series

Nowadays, huge quantities of satellite images are available from many different Earth observation sites. Moreover, thanks to a growing number of satellite sensors, the acquisition frequency of a same scene is permanently increasing. Furthermore, the high spatial resolution of the sensors gives access to detailed image structures. Thus, opportunities to compose high-resolution satellite image time-series (SITS) are growing and the observation of precise spatio-temporal structures in dynamic scenes is getting more and more accessible.

Experiments presented in this paper were performed using a SITS, partially visible in Fig. 1. The SITS is composed of SPOT multispectral images containing 2000 × 3000 pixels. The spatial resolution is 20 m. The acquired scene is a rural area located in the East of Bucharest (Romania). The acquisition campaign was driven in order to provide remote sensing data for the *Data Assimilation for Agro-Modeling (ADAM)* project. The SITS was obtained by daily acquisition and by filtering out images presenting a cloud or a snow cover above the project test sites. This selection procedure resulted in 38 images irregularly sampled in time, which were acquired over a period of 286 days. Fig. 2 displays the irregular sampling of the SITS. The images were then made superposable and a radiative transfer model was applied to produce reflectance measurements. The ADAM project SITS is available online.[1]

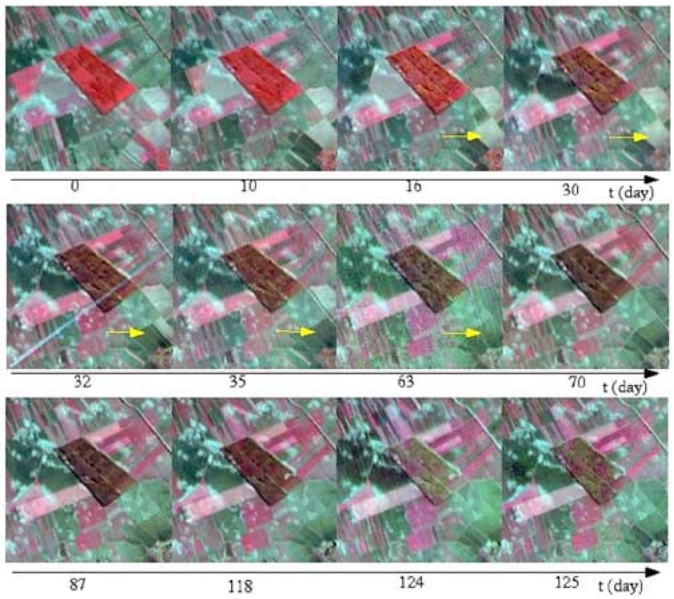

Fig. 1. SITS contain many spatio-temporal structures. The yellow arrow points out a ploughing phenomenon occurring in the ADAM SITS. We can also see a plane occlusion in the image acquired 32 days after the first acquisition (November 14, 2001), as well as the smooth evolution of the forest cover.

SITS are complex objects possessing a rich information content. They contain numerous and various spatio-temporal structures. For example in rural scenes, one can observe the growth and the maturation of cultures, their harvests, evolutions of ploughland, river floods, etc. Near urban areas, car and plane occlusions are frequent but there are also evolving constructions, pollution phenomenon, etc. Some examples of spatio-temporal structures are pointed out in Fig. 1.

The analysis of spatio-temporal structures are useful to understand complex evolutions which concern various domains such as agriculture, forest monitoring, ecology, hydrology, urbanization, etc. But our capacity to store these large volumes of data has exceeded our ability to access the broad variety of information contained in it. Indeed, limited tools exist to exploit this huge potential of information. Change detection, monitoring and validation of physical models by data assimilation constitute the most used analysis for information extraction in SITS.

Manuscript received August 14, 2003; revised December 9, 2004. This work was done in the framework of a CNES–DLR cooperation, in the field of information extraction and scene understanding from Earth observation data.

P. Héas is with the Laboratoire d'Informatique et Mathématiques Appliquées, Institut de Recherche en Informatique de Toulouse, 31071 Toulouse, France (e-mail: patrick.heas@enseeiht.fr).

M. Datcu is with the Remote Sensing Technology Institute (IMF), German Aerospace Center (DLR), D-82234 Wessling, Germany.

Digital Object Identifier 10.1109/TGRS.2005.847791

[1]Centre National d'Etudes Spatiales, database for the data assimilation for Agro-Modeling (ADAM) Project: http://medias.obs-mip.fr/adam/web/frameset/f_serveur.html.

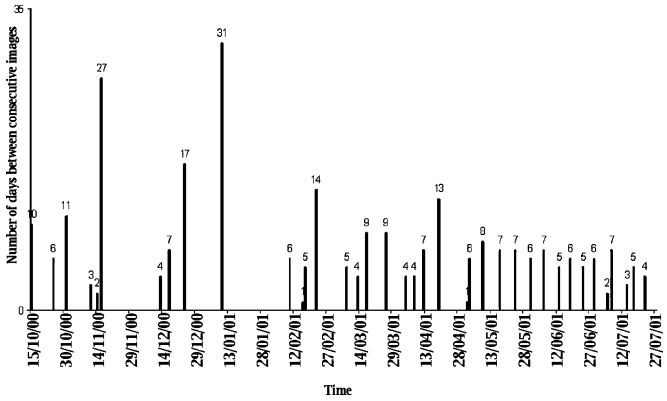

Fig. 2. Acquisition dates of cloud-free images in the ADAM database. The horizontal axis represents time which is irregularly sampled while the vertical axis represents the time difference between consecutive acquisition dates.

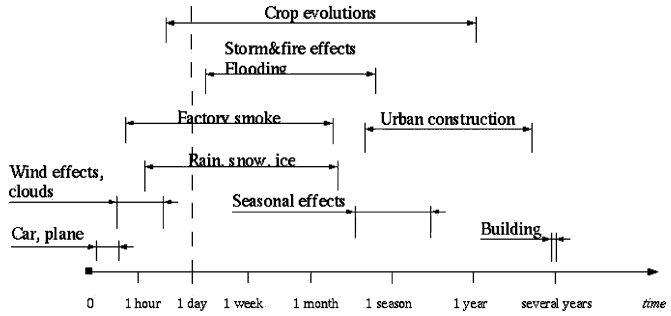

Fig. 3. Phenomenon process changes which occur in a dynamic scene have different time-scales. For example, plane occlusions are evanescent objects over short time-periods whereas buildings are stationary objects over long time-periods. Phenomena process which are changes relevant to crop evolutions, such as the growth of crops or their harvests, possess various time-scales. Spatio-temporal objects with time-periods below the blue dashed line, might possess higher frequencies in their spectrum than the sampling limit frequency.

The methods developed in these fields are complicated and dedicated to specific applications. Although these techniques are efficient, together they represent a limited range of applications. Nevertheless, one may be interested in finding a specific forest cover evolution or in detecting wheat harvests occurring during a given period. Until now, only few methods have been developed. They mainly focus on low-resolution images regularly sampled in time such as meteorological data [1], [2].[2] Thus, in order to adapt to a broader range of application and to have access to the variety of information contained in SITS, collaborative and generic methods are needed.

B. Information Mining

Large volumes of data are important resources. But to be relevant, users must be able to interpret the data information content. Understanding this huge quantity of data, which may be complex and multidimensional, can represent laborious work for users. Images are particularly complex objects possessing rich information contents. A manual analysis of associations and relations among images is not feasible. Furthermore, the usefulness of such an analysis may be restricted to a particular application. But there is a broad diversity of application domains and it is not possible to produce a specific analysis for each one of them. Generic analysis methods are needed to respond specifically to the needs of each application domain.

In summary, in many fields, there is a real need to transform growing databases into knowledge. The objective of information mining is to solve this problem, by adapting the data information content to the users' needs. Information mining can be defined as the nontrivial process of analyzing data in the perspective of discovering implicit but potentially useful information. The discovered information can be for instance patterns, association rules, causal effects, changes, anomalies, etc. An information mining perspective enables content-based retrieval, knowledge discovery, and data understanding.

The information mining problematic can be understood as a communication channel problem with: on one hand the database representing the information source and, on the other a user representing the receptor. Along the channel, the data are hierarchically processed, inducing a signal representation followed by a semantic representation. The signal representation is obtained by extracting information from the data by stochastic modeling of the signal. In such a case, extracted information is described with a particular model vocabulary, which is unmeaningful for users. The semantic representation is obtained by modeling the users' semantics. In such a case, information is described with a vocabulary and a syntax natural to users.

The goal of information mining is to bridge the semantic gap, that is to say, to minimize the loss of information between information available through the semantic representation, and information obtained by a direct data inspection.

Before going into the kernel of this paper which is the description of a SITS information mining concept, let us motivate our approach by pointing out some difficulties for SITS modeling.

The analysis of spatio-temporal structures in SITS is particular. Indeed, structures are characterized by heterogeneous temporal and spatial scales. Fig. 3 illustrates the variety of temporal scales attached to structures. Spatial scales of structures are also very different. Thus, SITS modeling methods should capture information at various scales. However, a pixel-localized time-series analysis is generally not appropriate to characterize high-resolution SITS structures. For the ADAM database in particular, the superposability difficulties, the irregular sampling, and the sampling limit frequency, prevent a pixel-localized stochastic modeling. The dashed vertical line in Fig. 3 illustrates this limit. It discriminates objects which possess higher frequencies in their spectrum than the sampling limit frequency. To fight against these constraints, analysis at an object level may be more robust against noise and superposability errors. Moreover, it may enable an investigation below the frequency limit by using contextual information. For instance, the behavior of the smoke of a factory, which is an object evolving in space according to the wind effects and in time according to the factory activities, can be modeled in a more efficient way using its spatial context rather than a space localized time-series analysis.

[2]See also National Aeronautics and Space Agency—Jet Propulsion Laboratory, literature database on ocean surface topography from space: http://topex-www.jpl.nasa.gov/science/time-series-data.html.

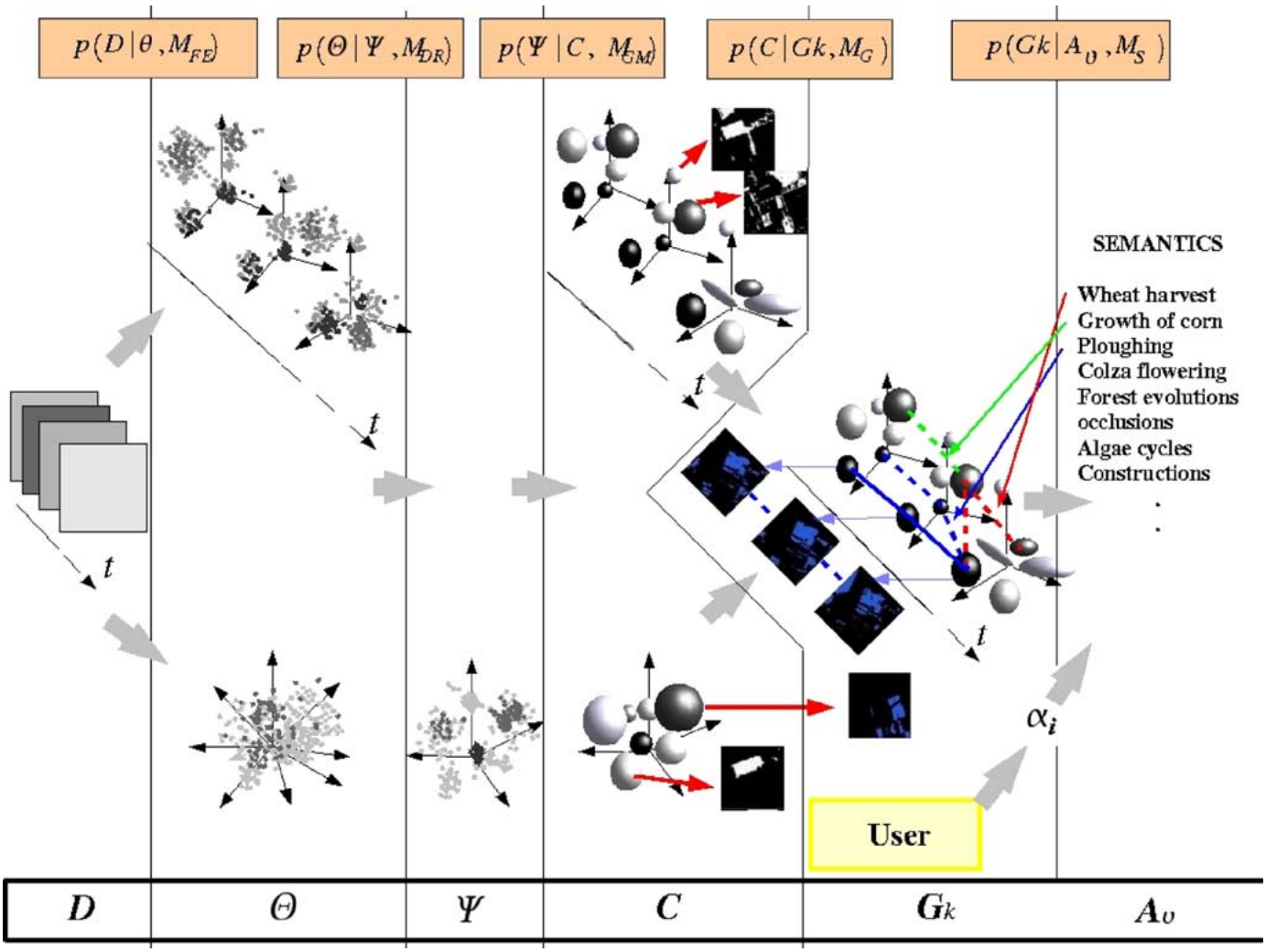

Fig. 4. Hierarchical modeling of SITS information content. The hierarchy enables users to link spatio-temporal structures to their specific interests. First, primitive SITS features $\boldsymbol{\theta}$ are extracted from the data $\boldsymbol{D}$. Two complementary representation are induced by using the TL feature spaces and the MT feature space. Next, dimension reduction techniques are applied and result in projected features $\boldsymbol{\psi}$. Then, the feature distributions are learned and clusters $\boldsymbol{C}$ and classes are discriminated. The unsupervised learning procedure is finally achieved by inferring graphs $\boldsymbol{\mathcal{G}_k}$ and dynamic classifications, which code the data structures. By interactive learning, the user interests $\boldsymbol{\mathcal{A}_\nu}$ are linked to the graphs and semantic labels are assigned to spatio-temporal structures.

To respond to the problematic of information mining in SITS, we presents in the following section a Bayesian hierarchical modeling of SITS information content. The different inference steps of the hierarchy are hereby detailed: we present in Section III an unsupervised learning procedure which results in a graph of dynamic cluster trajectories, and in Section IV we detail an interactive learning procedure which results in a semantic labeling of spatio-temporal structures. The graph of trajectories and the semantic labels constitute the signal and the semantic representations of the SITS information content. In Section V, experiments performed on the ADAM dataset are presented and finally, in the last section, conclusions are drawn.

II. Bayesian Hierarchical Modeling of SITS Information Content

In order to build an information mining system for dynamic scene understanding which is free from the application specificity and which enables its open use in almost any scenario, we use a Bayesian hierarchical model made up of six different levels (Fig. 4). The model links the information source $\boldsymbol{D}$, which represents a SITS comprising spatio-temporal structures, to the different users' semantics $\boldsymbol{\mathcal{A}_\nu}$. The hierarchy is defined by the relation between the random variables

$$D \rightarrow \Theta \rightarrow \Psi \rightarrow C \rightarrow \mathcal{G}_k \rightarrow \mathcal{A}_\nu \tag{1}$$

where $\boldsymbol{D}$, $\boldsymbol{\theta}$, $\boldsymbol{\psi}$, $\boldsymbol{C}$, $\boldsymbol{\mathcal{G}_k}$, and $\boldsymbol{\mathcal{A}_\nu}$ are the random variables realizations which are associated to more and more refined levels of information representation. As it is a Bayesian hierarchical model, the inference of a higher level in the hierarchy depends on the adjacent lower level and, conditionally to the latter, is independent of all other lower levels. Thus, given some data, we infer the most likely model by maximizing the joint distribution

$$p(\boldsymbol{D}|\boldsymbol{\theta})p(\boldsymbol{\theta}|\boldsymbol{\psi})p(\boldsymbol{\psi}|\boldsymbol{C})p(\boldsymbol{C}|\boldsymbol{\mathcal{G}_k})p(\boldsymbol{\mathcal{G}_k}|\boldsymbol{\mathcal{A}_\nu})p(\boldsymbol{\mathcal{A}_\nu}). \tag{2}$$

Hence, the levels of the hierarchy are learned independently by applying Bayesian inferences or inferences based on entropic measurements on families of stochastic models. Moreover, for each learning step, we can incorporate prior knowledge by using Bayes rule. We decompose the hierarchy into two parts:

- *unsupervised learning* $D \rightarrow \mathcal{G}_k$*:* for a particular SITS realization $\bar{\boldsymbol{D}}$, we infer a collection of graphs $\boldsymbol{\mathcal{G}_k}$ representing dynamic cluster trajectories coding spatio-temporal structures;
- *interactive learning* $\mathcal{G}_k \rightarrow \mathcal{A}_\nu$*:* the collection of graphs $\boldsymbol{\mathcal{G}_k}$ are linked to a collection of users' semantics $\boldsymbol{\mathcal{A}_\nu}$; thus we induce a semantic labeling of spatio-temporal structures of SITS.

The graph inference, which requires a significant computational cost, is an application-free learning procedure. Based on this objective representation, the semantic labeling, which is interactive, learns user-specific interests using positive and negative examples.

Before detailing the levels of the hierarchy, let us introduce several spaces for the SITS representation. Image time-series are stochastic processes which are usually represented in a multidimensional space comprising two spatial axes r, a time axis t, and several feature components θ (radiometric values, texture parameters, etc.). Since the features are assumed independent, the analysis of the multidimensional space is done independently on the different types of features. In such a multidimensional space, the signal denoted by $x(r,t,\theta)$ is represented by a cloud of points. This is the natural space for the SITS representation. But, SITS possess several other representations which lead to various interpretations. We hereby introduce as follows, several spaces for the SITS representation which are used in the proposed information mining concept.

The space formed by the feature components θ and the time axis t is called the *dynamic feature space*. In this space, SITS is represented by a histogram of features evolving in time. As the evolving features depend on their spatial location, we note as $x_r(t,\theta)$ this evolving histogram.

Considering d time samples, we denote by $\{\theta_{t_i}\}$ the d feature components localized at different times $\{t_i; i \in [1,\ldots,d]\}$. We can form d different *time localized (TL) feature spaces* with the d different time localized feature components θ_{t_i}. In them, we represent SITS by a succession of histograms of features. As each feature is linked to a spatial location, we note these signals as $\{x_r(\theta_{t_i}); \in [1,\ldots,d]\}$. These spaces constitute the different states of the dynamic feature space. If we group features by similarity, we obtain d different collections of clusters $\{C^j_{t_i}; j = 1,\ldots,K_{t_i}\}$. In the image space, we represent the d spatial classifications $\{C^j_{t_i}(r); j = 1,\ldots,K_{t_i}\}$ associated to the collections of TL clusters.

The *multitemporal (MT) feature space* is a multidimensional space composed of the d TL feature components $\{\theta_{t_i};\ i = 1,\ldots,d\}$. In this space we represent SITS by a multidimensional histogram of features. As this histogram depends on spatial locations, we note it as $x_r(\theta_{t_1},\ldots,\theta_{t_d})$. If we group features by similarity, we obtain a collection of clusters $\{C^j_{MT}; j = 1,\ldots,K_{MT}\}$. In the image space, we represent the spatial classification $\{C^j_{MT}(r); j = 1,\ldots,K_{MT}\}$ associated to the collection of MT clusters. By projecting the MT clusters in the different TL feature spaces, we can decompose the MT representation and reconstruct the different states of the dynamic feature space. We denote by $C^j_{MT_{t_i}}$, an MT cluster projected in the TL feature space at time t_i.

Equipped with these SITS representations, we present the different levels of the hierarchy.

- The lowest level represents the data $\boldsymbol{D}$, which is constituted by spatio-temporal structures defined in time windows and spatial masks.
- First, by using different signal models, features $\boldsymbol{\theta}$ are extracted from the data at a pixel level for the different time locations $\{t_i\}; i = 1,\ldots,d$. We then induce two complementary representations by placing them in the TL feature spaces and in the MT feature space.
- Next, to fight against the "curse of dimensionality," we employ dimension reduction techniques to extract, from the MT feature space, interesting projections containing linear and nonlinear structures. The projected features $\boldsymbol{\psi}$ are represented in a space composed of the extracted components.
- Then, the distribution of the MT projected features and the TL features are learned using a Gaussian mixture model of unknown complexity. The modeling procedure discriminates MT and TL clusters $\boldsymbol{C}$ possessing Gaussian shapes. MT and TL classifications are then produced, by mapping these clusters in the image space.
- The unsupervised learning procedure is finally achieved by inferring graphs $\boldsymbol{\mathcal{G}_k}$ coding the data structure of SITS. They model the dynamic feature space by formalizing trajectories of MT clusters through TL clusters. Additionally, a spatial constraint is introduced into the inference by using the MT and TL classes.
- By interactive learning, the users' interests $\boldsymbol{\mathcal{A}_\nu}$ are linked to the graphs which represent spatio-temporal structures. To complete this semantic labeling, parameters of a graph similarity model are interactively estimated by updating probabilities of a Bayesian network. This update is performed using a Dirichlet model with positive and negative examples provided by a user.

III. Unsupervised Inference of a Graph of Dynamic Cluster Trajectories

A. Feature Extraction

Dynamic scene understanding relies on the ability and robustness of information extraction from the observed data. We apply appropriate stochastic models M_{FE} to capture spatial, spectral, or geometric structures in each image of the time-series at a pixel level. These models are given as parametric data models $p(\boldsymbol{D}|\boldsymbol{\theta}, M_{FE})$ and assign a likelihood to a given realization of the data $\bar{\boldsymbol{D}}$ for a particular value of the parameter vector $\boldsymbol{\theta}$. Examples of these image models are Gibbs–Markov random field models for textural features or the intensities of the multispectral images for spectral features [3]. Of course, for the latter, no sophisticated modeling is involved.

The extracted features are represented in the MT feature spaces by the signal $x_r(\theta_{t_1},\ldots,\theta_{t_d})$. Together with the implicit spatial index r, this signal carries the global information on the time-series. But, these extracted features can also form, in the TL feature spaces, a collection of signals $\{x_r(\theta_{t_i})\}$ with $i = 1,\ldots,d$.

In the next two sections, we detail the analysis of the extracted features represented first in the MT feature space and then in the

TL feature spaces. Finally, we use these two types of representation to model the dynamic feature space.

B. Modeling a Multitemporal Feature Space

1) Dimension Reduction: The MT feature space represents a space of high dimensionality since it results from the union of all the TL feature spaces. A direct application of a nonparametric procedure is severely restricted in this case, by the limitation called the "curse of dimensionality" [4]. However, the information contained in a feature space of high dimensionality can often be represented with fewer dimensions. In remotely sensed images in particular, the spectral bands usually present redundancies. Furthermore, the phenomenon is likely to be accentuated when considering an MT feature space. Dimension reduction techniques, exploit this property to reduce the space dimension by extracting interesting projections.

Dimension reduction techniques study statistical methods like principal component analysis (PCA) or projection pursuit (PP), which are to some extend, equivalent to methods based on discriminant analysis [5]. Indeed, PCA is suited for linear analysis while PP can exploit the extra compression associated with nonlinear relationships. Furthermore, the application of these statistical methods, enables us to release from the singularity problems which often occur in high-dimensional spaces. Resulting projections are then modeled using a Gaussian mixture assumption (cf. Section III-B2) in which multimodal patterns are discriminated.

Assuming a model M_{DR} for the m-dimensional distribution associated with n feature realizations $\boldsymbol{\theta} = \{\theta^1, \ldots, \theta^n\}$, the quality of the l-dimensional projected features $\boldsymbol{\psi} = \{\psi^1, \ldots, \psi^n\}$ can be evaluated by the likelihood $p(\boldsymbol{\theta}|\boldsymbol{\psi}, \mathcal{M}_{DR})$. Gaussian distributions for PCA or just non-Gaussian distributions for PP are examples of these models.

Principal component analysis is a linear projection of an m-dimensional space into a space in which the axes of the projections called principal component axes are decorrelated. Moreover, the principal components are ordered according to a variance criterion. In other words, the jth eigenvector of the data covariance matrix corresponding to its jth biggest eigenvalue defines the jth principal component axis. This eigenvalue decomposition is more convenient than the maximization of the projection likelihood. The analysis relies on the assumption that the data has a normal distribution in the feature space.

From another perspective, PCA searches for an orthogonal base which minimizes the quadratic distance between the vectors $\{\theta^i;\ i = 1, \ldots, n\}$ of the m-dimensional data and their orthogonal projection on this base. This is equivalent to maximizing the inertia of the projection defined by

$$I = \frac{1}{n} \sum_{i=1}^{n} \|\psi^i\|_M^2. \tag{3}$$

The index M here outlines the fact that the data have been normalized by an appropriate matrix M and transformed in order to have a mean equal to zero. In order to perform a dimension reduction, only l principal components with $l < m$ should be selected. To evaluate the loss of information, we use the signal energy $\mathcal{W} = \sum_{i=1}^{m} \lambda_i$ where the λ_i represent the eigenvalues of the features $\boldsymbol{\theta}$ autocorrelation matrix. Thus, selecting the l first principal components corresponds to a percentage α of restitution of the signal energy equal to

$$\alpha = \frac{\sum_{i=1}^{l} \lambda_i}{\mathcal{W}} * 100. \tag{4}$$

Projection pursuit groups dimension reduction techniques that extract linear projections which contain nonlinear structures from a multidimensional space. Furthermore, under certain assumptions, the extracted components are independent from the others. The extracted components are ordered according to a criterion of non-Gaussianity evaluated by a projection index. This approach is equivalent to the maximization of the projection likelihood [6]. The analysis rely on the very general assumption that the data possesses a non-Gaussian distribution, which is in most cases, a valid hypothesis. After selecting a sufficiently large percentage of the signal energy with PCA, we apply PP on the reduced signal in order to exploit the extra compression associated with nonlinear relationships.

The higher the dimensionality of the extracted projections, the richer their information contents. For example, structures in a plan can not always be detected in one-dimensional projections. We perform our analysis using a bidimensional projection index based on the chi-square distance. The relevance of this distance for the approximation of Kullback–Leibler divergence has been shown in [7]. In order to reveal all the maxima of the projection index, we employ an efficient stochastic optimization procedure proposed by Posse [8] combined with the structure removal technique proposed by Friedman [9].

To perform a dimension reduction, the l first independent components with $l \leq m$ are selected. P-values are employed in order to decide whether the components are the effect of noise or are really independent. Indeed, P-values are useful to determine limits which correspond to probabilities of projection independence [7]. Thus, for a given index limit called quantile, l independent dimensions are extracted and dimension reduction is achieved.

Dimension reduction techniques such as PCA and PP are able to condense the information contained in the MT feature space into a subspace of lower dimensionality. They provide an efficient solution for parameter and model inference in multidimensional spaces with limited sample sizes.

2) Mixture Modeling:

Gaussian mixture models: Stochastic models are appropriate tools to learn about this multidimensional signal. A Gaussian mixture model is able to approximate efficiently, any distribution for which no prior knowledge is available and in particular multimodal distributions. Thus, a Gaussian mixture assumption is well suited to model the distribution of the l-dimensional realizations $\boldsymbol{\psi} = \{\psi^1, \ldots, \psi^n\}$ of the random variable Ψ, which are assumed independent and identically distributed. A mixture modeling procedure can infer similarities that can then be used for clustering the feature space. Components of the mixture are constituted by the grouping of similar feature points and thus, will define clusters $\boldsymbol{C}$.

For K components weighted by $\{\pi_k\}$ and possessing mean vectors noted as $\{M_k\}$ and covariance matrices noted as $\{A_k\}$,

the l-dimensional distribution of a Gaussian mixture model M_{GM} is defined by

$$p(\psi^i|C, M_{GM}) = \sum_{k=1}^{K} \pi_k p(\psi^i|M_k, A_k), \qquad \sum_{k=1}^{K} \pi_k = 1 \quad (5)$$

where

$$p(\psi^i|M_k, A_k) = \frac{\left|A_k^{-1}\right|^{\frac{1}{2}}}{(2\pi)^{\frac{l}{2}}} \cdot \exp\left\{-\frac{1}{2}(\psi^i - M_k)^T A_k^{-1}(\psi^i - M_k)\right\}. \quad (6)$$

In order to perform the modeling procedure without any constraints either on the number of Gaussians present in the mixture or on their parameters, a criterion is needed to select the best model among all the possible Gaussian mixture configurations. In order to infer among a collection of models, the minimum description length (MDL) principle is applied.

Model selection by the MDL principle: For the n realizations $\boldsymbol{\psi} = \{\psi^1, \ldots, \psi^n\}$, we choose out of a finite set of possible models $\{\mathcal{M}_1, \ldots, \mathcal{M}_M\}$, a model hypothesis $\mathcal{M}_j$ of distribution $p(\boldsymbol{\psi}|\mathcal{M}_j)$ for $\boldsymbol{\psi}$. We consider also the code length function (measured in bits) $L_{\mathcal{M}_j}(\boldsymbol{\psi})$ needed for the description of $\boldsymbol{\psi}$ under the model hypothesis $\mathcal{M}_i$. A bijection appears between the probability distribution $p(\boldsymbol{\psi}|\mathcal{M}_j)$, and the code length function $L_{\mathcal{M}_j}(\boldsymbol{\psi})$. Indeed, $\forall \boldsymbol{\psi}$

$$L_{\mathcal{M}_j}(\boldsymbol{\psi}) = -\log_2 p(\boldsymbol{\psi}|\mathcal{M}_j). \quad (7)$$

Minimizing this code length, called by Rissanen, "stochastic information complexity," over $\mathcal{M}_j$ selects the model maximizing the Bayesian evidence. But the computational cost of this quantity is often prohibitive. A first-order approximation is achieved by the so-called "two-part MDL code" [10]. For parametric model families, this code length function noted as $L_{2P}(\boldsymbol{\psi})$ is composed of two terms: the code length necessary to encode the model and its estimated parameter $\widehat{\phi}$, and the code length necessary to encode the data keeping in mind the model and its estimated parameters. The first description length part is induced by the model and parameter encoding using noninformative prior distributions. It is noted as $-UnivCode(\mathcal{M}_j)$. The second description length part is related to the model maximum likelihood $p(\boldsymbol{\psi}|\widehat{\phi}, \mathcal{M}_j)$. Thus, the two-part MDL code is defined by

$$L_{2P}(\boldsymbol{\psi}) = -\log_2 p(\boldsymbol{\psi}|\widehat{\phi}, \mathcal{M}_j) - UnivCode(\mathcal{M}_j). \quad (8)$$

The MDL principle states that the best model among a collection of tentatively suggested ones, is the one that encodes the data with the smallest code length. To estimate the code length, we can use the stochastic information complexity or its first order approximation, the two-part MDL code. The computation of this approximation requires a less intensive calculation procedure and is particularly convenient for mixture modeling.

MDL principle for Gaussian mixture modeling: On the basis of the two-part MDL code, we derived the description length of the data for the family of Gaussian mixture models. A simplified model, neglecting the influence of surrounding Gaussian components and assuming constant variances for the Gaussians, was previously developed in [11]. A two-part description length, derived from modeling a mixture of uncorrelated Gaussians, has been proposed by Wallace and Dowe [12]. In this paper, we extend this algorithm to the correlated Gaussian mixture model. This algorithm is to some extend, equivalent to the Bayesian Autoclass algorithm [13]. We assume the hypothesis of noninterfering Gaussians. The two-part MDL code length for encoding the data $\boldsymbol{\psi} = \{\psi^1, \ldots, \psi^n\}$ using a Gaussian mixture model of K Gaussians of dimensionality l, is defined by (9), shown at the bottom of the page, where $K_{\max}$ denotes a maximum number of Gaussians in the mixture and G_k denotes the kth Gaussian of the mixture. Appendix I details how this two-part description length is derived.

Optimization: The goal is to estimate K, $\{(M_k, A_k)\}$ and $\{\pi_k\}$, by minimizing $L_{2P}(\boldsymbol{\psi})$. Enumerating all configurations and evaluating the two-part MDL code is not feasible. Instead, an optimization algorithm which evaluates the changes of the code length between two configurations rather than the code length itself is used.

Before introducing the general algorithm, one can prove that the change of the code length induced by the removal of a given Gaussian $G_{k'}$ is defined by (10), shown at the bottom of the page, where π_j and $\pi_{jk'}$ are the maximum-likelihood (ML) estimates of the weight of clusters j before and after removal of $G_{k'}$, and where $p_j(.)$ and $p_{jk'}(.)$ are respectively the ML estimated probability distribution of G_j before and after removal of $G_{k'}$. $p_{k'k'}(\psi_i)$ is the ML estimated distribution of the most probable Gaussian which is assigned to ψ_i after removal of $G_{k'}$. Appendix II provides a proof for (10).

The optimization algorithm is composed of the following steps:

$$L_{2P}(\boldsymbol{\psi}) = \log_2(K_{\max}) + K\log_2(n) + \sum_{k=1}^{K}\left(\frac{l(l+1)}{2}\log_2(n\pi_k) - n\pi_k \log_2 \pi_k + \sum_{\psi_i \in G_k}\left(\frac{1}{2}\log_2 \frac{\left|A_k^{-1}\right|}{2\pi^l} - \frac{{}^t(\psi^i - M_k)A_k^{-1}(\psi^i - M_k)}{ln(2)}\right)\right) \quad (9)$$

$$\Delta_{G_{k'}}(L_{2P}(\boldsymbol{\psi})) = -\log_2(n) - \frac{l(l+1)}{2}\log_2(n\pi_{k'}) + n\pi_{k'}\log_2 \pi_{k'} - \sum_{j=1, j\neq k'}^{K} n\left((\pi_j + \pi_{jk'})\log_2(\pi_j + \pi_{jk'}) - \pi_j \log_2 \pi_j\right) + \sum_{\psi_i \in G_{k'}} \log_2 \frac{p_{k'}(\psi_i)}{p_{k'k'}(\psi_i)} + \sum_{j=1, j\neq k'}^{K} \sum_{\psi_i \in G_j} \log_2 \frac{p_j(\psi_i)}{p_{jk'}(\psi_i)} \quad (10)$$

Step 1) *Initialization*: A initial Gaussian mixture is produced. It is composed of a high number $K^{(0)}$ of Gaussian, with parameters noted as $\{(M_k^{(0)}, A_k^{(0)})\}$ and $\{\pi_k^{(0)}\}$. The initialization is done by randomly spreading the clusters according to a Gaussian distribution of mean and variance learned from each data feature component.

Step 2) *Adaptation*: At iteration (q), we consider $K^{(q)}$ Gaussians in the mixture. An expectation–maximization (EM) algorithm [14] is used to perform a ML estimation of the Gaussian mixture parameters $\{(\widehat{M}_k^{(q)}, \widehat{A}_k^{(q)})\}$ and $\{\widehat{\pi}_k^{(q)}\}$.

Step 3) *Selection*: For the same iteration (q), we remove the Gaussian $G_{k'}$ which induces the biggest decrease in the description length $\Delta_{G_{k'}}^{(q)}(L_{2P}(\boldsymbol{\psi^n}))$. Then we increment (q) and go back to Step 2). If no decrease is observed, that is to say if $\forall G_{k'}$, $\Delta_{G_{k'}}^{(q)}(L_{2P}(\boldsymbol{\psi^n})) < 0$, then we do not remove any Gaussians and go to Step 4).

Step 4) *Convergence*: if at iteration (q_end), no other decrease in the description length is observed, then the algorithm stops iterating Steps 2) and 3). We then obtain the estimated number of Gaussians $\widehat{K} = K^{(q\_end)}$ with the ML estimates of the parameters of the mixture model $\{\widehat{M}_k^{(q\_end)}, \widehat{A}_k^{(q\_end)}, \widehat{\pi}_k^{(q\_end)}\}$.

The MT feature space is modeled according to a Gaussian mixture distribution. Thus, we learn the parameters K_{MT}, $\{(M_k, A_k)\}$ and $\{\pi_k\}$ related to the Gaussian mixture model. The modeling procedure infers similarities which are then used to cluster this multidimensional feature space. Therefore, each Gaussian G_k comprises feature points and defines a cluster C_{MT}^j. In parallel, MT spatial classes $C_{MT}^j(r)$ of the image time-series are generated.

C. Time-Localized Representations

Our interest is focused on understanding the dynamic of image structures in different time locations. To achieve this goal, there are two different approaches to decompose the signal in time.

1) Projecting the Multitemporal Feature Space: From an initial perspective, the signal representation in the MT feature space $x_r(\theta_{t_1}, \ldots, \theta_{t_d})$ can be projected to enable TL representations. Consequently, each MT cluster C_{MT}^j with $j \in [1, K_{MT}]$ can be projected into d different TL feature spaces. We obtain projected MT clusters denoted by $\{C_{MT_{t_i}}^j; t_i \in [1, d]\}$. The projected clusters are representative of global behaviors decomposed in time. Furthermore, they are specific of the MT feature space modeling. Consequently, they contain information about the time evolution of the feature distribution.

2) Modeling the Time-Localized Feature Space: New modeling procedures can be performed directly for each of the TL representations $x_r(\theta_{t_i})$, independently from the MT feature space modeling. This procedure produces d sets of TL clusters $\{C_{t_i}^j; t_i \in [1, d]\}$ with $j \in [1, K_{t_i}]$, where K_{t_i} is the number of estimated clusters at time t_i. In parallel, spatial classes $C_{t_i}^j(r)$ are obtained. In this case, the TL clusters are defined for given time locations which are specific to the TL feature space modeling. Consequently, they contain TL information on the feature distribution.

To perform these d TL clusterings, we use the MDL-based Gaussian mixture modeling algorithm defined in Section III-B2.

3) Complementarity of the Representations: The MT feature space contains global information including the TL information. Moreover, for TL clustering, the separability of the different clusters is not as clear as for the MT case. However, TL analysis in contrast to the analysis of the highly dimensional MT feature space, allows a more detailed information extraction. Consequently, as the interest is a time decomposition of the signal, one should associate these two different TL representations for a complete understanding of dynamic clusters.

D. Modeling the Dynamic Feature Space

Our interest is now focused on modeling the signal $x_r(t, \theta)$ represented in the dynamic feature space.

Each MT cluster has a particular behavior when observing its evolution into successive TL representations. For example, some MT clusters may share the same cluster at a given time and split or/and merge with other MT classes at another time. The problem is to quantify, at a given time, the similarity of these projected MT clusters with the goal of inferring spatio-temporal relations. Since our interest is particularly time locations and according to the previous remarks on the complementarity of both TL representations, we propose the following model of trajectories: the trajectories of the projected MT clusters $C_{MT_{t_i}}^k$ at the different times $\{t_i; i = 1, \ldots, d\}$ are formalized using collections of TL clusters $C_{t_i}^j$.

1) Inference of Dynamic Cluster Trajectories: Based on these considerations, we define a model, noted as M_G, for the dynamic cluster trajectories. This model is a probabilistic distribution on the MT cluster collection $\boldsymbol{C}$ conditioned by a graph of trajectories $\boldsymbol{\mathcal{G}}$ constituted with TL clusters. Thus, we define the likelihood $p(\boldsymbol{C}|\boldsymbol{\mathcal{G}}, M_G)$ of a given a graph of trajectories.

To express this distribution, we hereby introduce a few notations. We decompose the graph in a set of K_{MT} graphs of trajectories $\boldsymbol{\mathcal{G}} = \{\mathcal{G}_1, \ldots, \mathcal{G}_{K_{MT}}\}$ formed by TL clusters, where each graph $\mathcal{G}_k$ is associated to an MT cluster $C_{MT}^k \in \boldsymbol{C}$. In order to decompose this joint probability distribution, we assume that the constitution of the trajectories associated to the different MT clusters are independent and that the association of an MT cluster with a given TL cluster is independent of associations with other TL clusters (localized at the same time but also at other times). Therefore, for a given MT cluster set $\boldsymbol{C}$, we derived a likelihood of a graph of trajectories $\mathcal{G}$ with the double product

$$p(\boldsymbol{C}|\boldsymbol{\mathcal{G}}, M_G) = \prod_{k=1}^{K_{MT}} \prod_{(i,j), C_{t_i}^j \in \mathcal{G}_k} p\left(C_{t_i}^j | C_{MT_{t_i}}^k\right). \quad (11)$$

To apply this model, we need to define the probabilities $p(C_{t_i}^j | C_{MT_{t_i}}^k)$. We use the Kullback–Leibler divergence [15] noted as $D(C_{MT_{t_i}}^k, C_{t_i}^j)$, which is an entropic measurement able to compare the two different TL distributions and learn about their similarity. Because of the relative interest for each

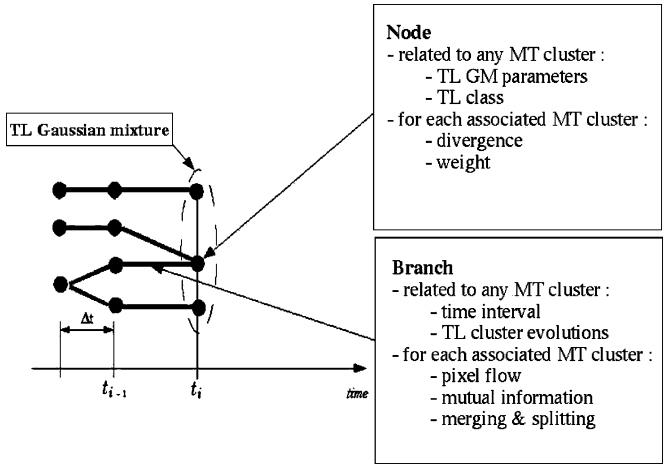

Fig. 5. Description of the graph of dynamic cluster trajectories.

of the MT class separately, a spatial constraint is introduced: this divergence measurement is weighted by the number of data points belonging to both of the corresponding MT and TL spatial classes. Let us denote by $card(.)$ an operator taking as an argument a spatial map and returning its population. Thus, we define the probabilities

$$p\left(C_{t_i}^j | C_{MT}^k\right) = \frac{1}{Z} \frac{card\left(C_{MT}^k(r) \cap C_{t_i}^j(r)\right)}{D\left(C_{MT_{t_i}}^k, C_{t_i}^j\right)} \tag{12}$$

where Z is a normalization constant. Note that the clusters are characterized by multivariate Gaussian distributions and thus, the divergence calculation is performed analytically. The maximum of the likelihood probability [cf. (11)] can be obtained by using graphs $\mathcal{G}_k$ comprising all the TL clusters $C_{t_i}^j$. But as the objective is to infer only the most likely associations of MT clusters with TL clusters, we limit the graphs complexity by removing associations with TL clusters which possess a probability $p(C_{t_i}^j | C_{MT}^k)$ below a given threshold μ. Thus, the graphs $\mathcal{G}_k$ maximizing the likelihood $p(\boldsymbol{C}|\boldsymbol{\mathcal{G}}, M_G)$ are simply those constituted with TL clusters for which probabilities $p(C_{t_i}^j | C_{MT}^k)$ are higher than μ.

Hence, using MT clusters, we infer graphs of trajectories of dynamic clusters which are composed of TL clusters and where the complexity of the graphs depends on a threshold parameter μ. These graphs constitute a model for the signal representation $x_r(t, \theta)$ in the dynamic feature space.

2) Graph of Dynamic Cluster Trajectories: The image time-series $x(r, t, \theta)$ has been previously submitted to several processing levels. They result in a specific representation $x_r(t, \theta)$ which is a graph modeling the trajectories of dynamic clusters. The chronology of the time-series and the irregular sampling information are stored in the graph. The trajectories information is condensed in the nodes and branches of the graph. Fig. 5 summarizes the description of the graph characterizing the dynamic clusters.

- A node represents a TL cluster $C_{t_i}^j$ defining the jth component of the Gaussian mixture at a given time t_i and is related to a collection of MT clusters by a set of probability measurements. The complete Gaussian mixture at a given time t_i is described by the entire set of TL clusters. Each MT cluster C_{MT}^k associated to the node $C_{t_i}^j$ is characterized by a pixel weight $card(C_{MT_{t_i}}^k(r) \cap C_{t_i}^j(r))$, a divergence measurement and TL Gaussian parameters. Moreover, each node regroups a set of indexed points in time and space represented in a TL class $C_{t_i}^j(r)$.
- The branches of the graph represent the MT cluster evolutions between two image samples. A branch, linking two consecutive TL clusters $C_{t_i}^j$ and $C_{t_{i-1}}^l$ which is related to a given MT cluster $\{C_{MT}^k\}$, is characterized by a time sampling interval Δt_i, a pixel flow and TL and MT Gaussian parameter evolutions. The flow of feature points exchanged between the TL cluster $C_{t_i}^j$ and $C_{t_{i-1}}^l$ is the number of pixels shared by the two TL spatial classes $C_{t_i}^j(r)$ and $C_{t_{i-1}}^l(r)$. Furthermore, because of the restriction to a given MT class, the flow is determined by the number of pixels shared by the previous TL spatial classes and the MT spatial class $C_{MT}^k(r)$. The quantization of flow of feature points enables us to evaluate the merging and the splitting of the dynamic clusters in time and in each feature dimension. These phenomena are simply related to the number of in going and out going branches associated to the nodes.

  The internal MT cluster changes between two consecutive times t_{i-1} and t_i can be quantified by mutual information. By using the projections of an MT cluster in two consecutive times t_{i-1} and t_i, mutual information between the two Gaussian projections can be measured using an analytical computation. Appendix III details the mutual information analytical calculation.

The graph characterizing the dynamic clusters $x_r(t, \theta)$ is a representation of the signal where the spatial variable r is hidden. However, spatial indexes related to each point in this feature space representation are accessible. Exploiting them permits us to generate representations in the image space. Indeed, we can associate to the K_{MT} different MT cluster trajectories, K_{MT} different representations in the spatio-temporal space. This space is formed by the spatial and temporal components r and t. We call these representations dynamic classifications. At each time $t_i \in [1, d]$, each dynamic classification is composed of a particular combination of TL classes $C_{t_i}^j(r)$. The TL classes of the dynamic classification related to the kth MT class $C_{MT}^k(r)$, are those which correspond to TL clusters in the associated graph of trajectories $\mathcal{G}_k$.

These dynamic classifications constitute the signal $x_\theta(r, t)$ which contains spatio-temporal information missing in the dynamic feature space representation $x_r(t, \theta)$. Together, these representations describe objectively the feature evolution and the spatial evolution of the image time-series $x(r, t, \theta)$.

IV. User-Specific Semantic Labeling by Interactive Learning

In this section, we focus on a very important step in providing content-based query techniques: the interaction with the user and the flexible incorporation of user-specific interests. It constitutes the last level of the hierarchical information modeling. The semantic modeling detailed in this section was previously

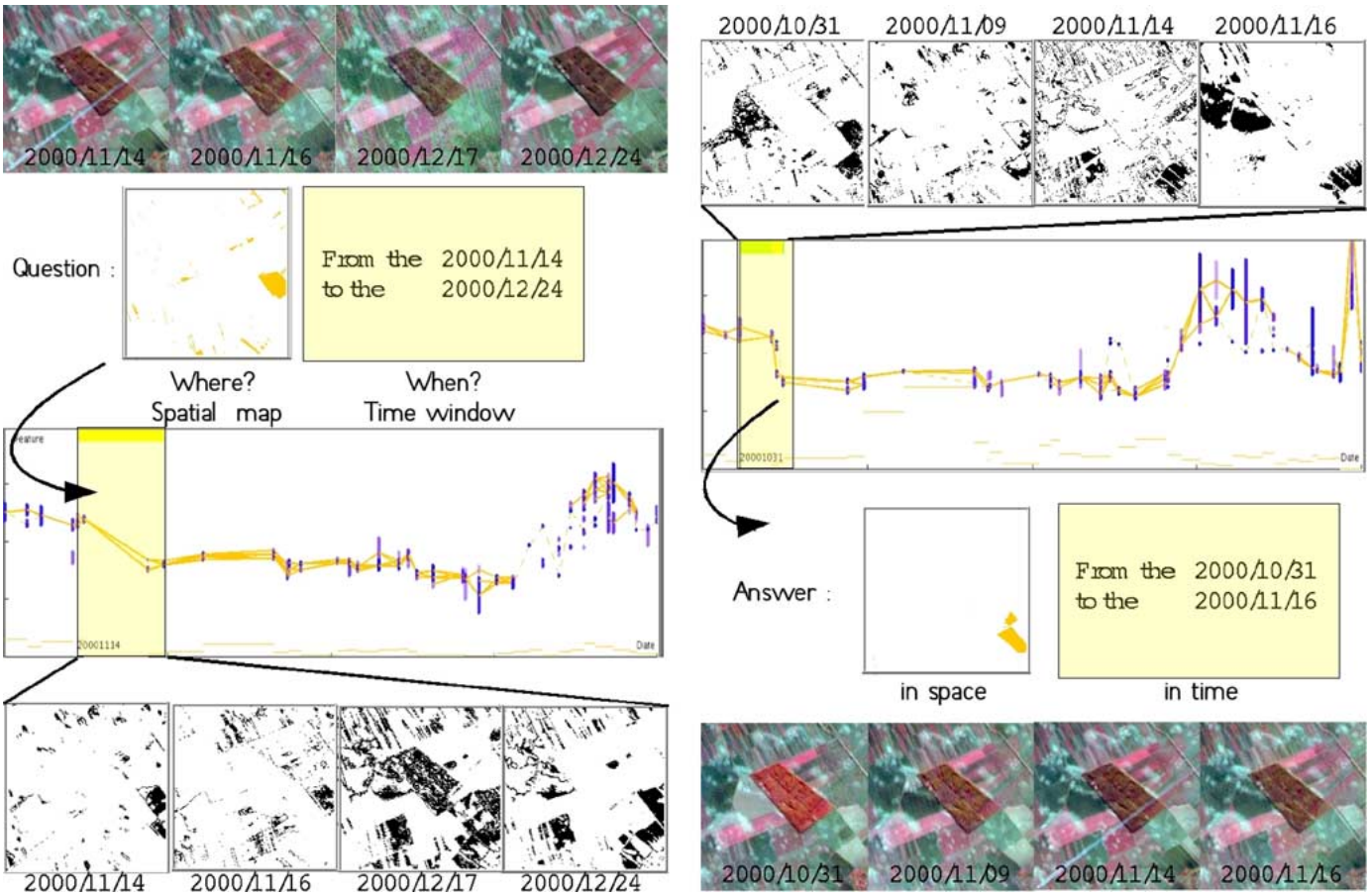

Fig. 6. Interactions between a user and a graph-based learning system. The user transmits time-windows and a spatial masks to the system. They correspond to spatio-temporal patterns of interest or noninterest; based on the graph representations of these examples associated to their dynamic classifications, the system learns interactively user-specific semantics and retrieves, in time and in space, similar spatio-temporal structures.

presented in [16]. The learning framework presents similarities with the one adopted by Schroder *et al.* [3].

Spatio-temporal processes, present at a given time and in a spatial window, can possess subjective user-specific semantics denoted by $\mathcal{A}_\nu$. A user may be interested in retrieving similar events and thus, may want to know when and where similar spatio-temporal patterns occurred. Moreover, the inference of the graph $\mathcal{G}$ is a robust and unsupervised coding of SITS. And, as subgraphs $\mathcal{G}_k$ contained in $\mathcal{G}$ are stochastic models for these spatio-temporal patterns, they can also possess user semantics.[3] Therefore, based on this objective signal characterization, we are interested in learning semantics from users in order to achieve a semantic labeling of subgraphs representing spatio-temporal patterns. Such a learning procedure could enable the recognition and the probabilistic retrieval of similar events.

In this perspective, we schematize in Fig. 6 the interactions between a user and a graph-based learning system: the user transmits to the system time-windows and spatial masks corresponding to spatio-temporal patterns of interest or noninterest; based on the graph representations associated to these positive and negative examples, the system learns interactively user-specific interests and retrieves a collection of spatio-temporal structures with similar semantics occurring in defined time-windows and spatial masks.

In order to define a model for user-specific semantics, a parametric similarity measure $S_\phi(\mathcal{G}_0, \mathcal{G}_k)$ between two subgraphs $\mathcal{G}_0$ and $\mathcal{G}_k$ is employed [16]. This measure is an extension of the inexact matching algorithm proposed in [17]. The parameter vector ϕ weights the contribution of each type of subgraph features. A given parameter vector corresponds to a particular user-specific similarity and formalize a particular semantic.

By defining interactively a similarity, it is possible to link the subjective elements $\mathcal{A}_\nu$ representing the user semantics to the objective subgraph features $\mathcal{G}_k$ by learning the likelihood

$$p(\mathcal{G}_k|\mathcal{A}_\nu, M_S) = \frac{S_{\widehat{\phi}}(\widehat{\mathcal{G}}_0, \mathcal{G}_k)}{S_{tot}} \tag{13}$$

where $\widehat{\phi}$ is an estimated parameter vector, $\widehat{\mathcal{G}}_0$ is an estimated reference subgraph, and S_{tot} is a normalization constant enabling to map the similarity function values into probabilities.

The estimation of this parameter vector is made interactively by updating the probabilities of a Bayesian network with user examples. More precisely, the probability update is performed indirectly by adjusting the hyperparameters vector $\alpha = \{\alpha_i\}$ of a Dirichlet model, depending on the users' examples. For further details, we refer the reader to [16].

[3]Conversely to notations of Section III-D1, graphs $\mathcal{G}_k$ are not necessarily associated to an entire MT trajectory, but can also be only parts of the MT trajectories.

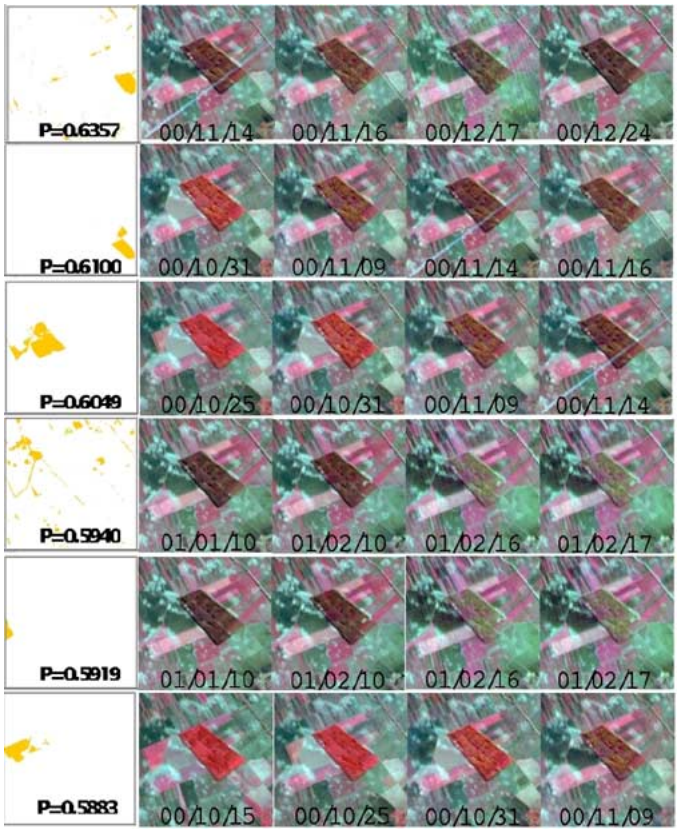

Fig. 7. Interactive learning of the ploughland semantics: most likely retrieved spatio-temporal structures ranked, from top to bottom, according to their posterior probabilities. Each row presents a retrieved spatial mask (left) with its associated time-period, which is given by time locations in the first and last images of the row. The images in each row show the spatio-temporal structure representations in SITS.

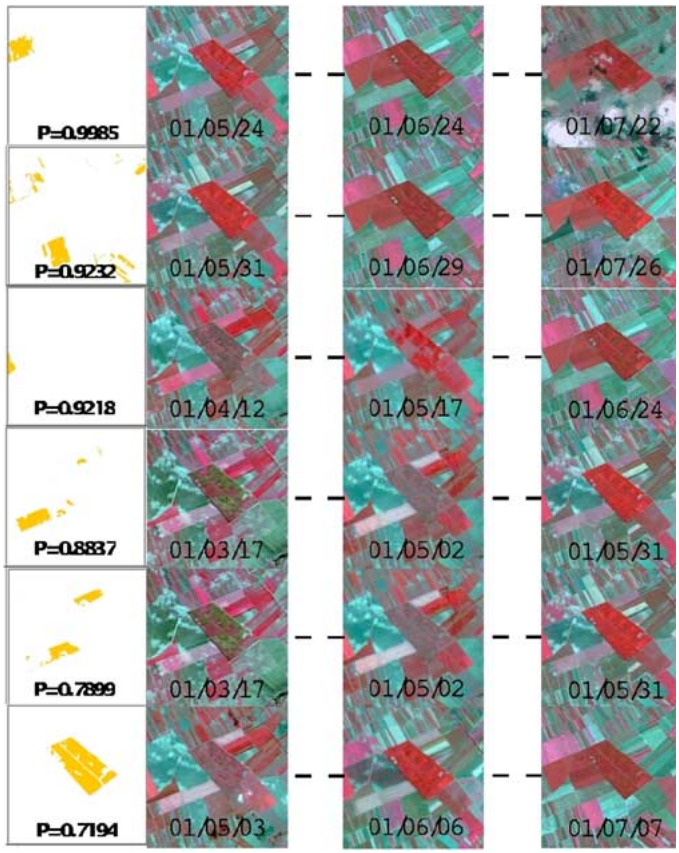

Fig. 8. Interactive learning of the field maturation semantics: most likely retrieved spatio-temporal structures ranked, from top to bottom, according to their posterior probabilities. Each row presents a retrieved spatial mask (left) with its associated time-period, which is given by time locations in the first and last images of the row. The middle images in each row were selected from SITS in order to represent significantly the spatio-temporal structure behaviors.

For notation simplification, the conditioning of the likelihood by a model M_S is omitted in the following. Based on the likelihood, using a Bayesian context enables the estimation of posterior probabilities $p(\mathcal{A}_\nu|\mathcal{G}_k)$ and thus, allows a semantic representation of the SITS content. Indeed, considering that a user provides positive or negative examples, corresponding to a positive $\mathcal{A}_\nu$ or negative $\neg\mathcal{A}_\nu$ semantics, two likelihood probabilities $p(\mathcal{G}_k|\mathcal{A}_\nu)$ and $p(\mathcal{G}_k|\neg\mathcal{A}_\nu)$ can be derived for each subgraph. Moreover, graph priors can be obtained using the formula $p(\mathcal{G}_k) = \sum_\nu p(\mathcal{G}_k|\mathcal{A}_\nu)p(\mathcal{A}_\nu)$, where the summation is done over the positive and negative semantics. Thus, assuming a uniform prior on the semantics, the posterior probability of the positive semantics is inferred using Bayes rule

$$p(\mathcal{A}_\nu|\mathcal{G}_k) = \frac{p(\mathcal{G}_k|\mathcal{A}_\nu)p(\mathcal{A}_\nu)}{p(\mathcal{G}_k)} = \frac{p(\mathcal{G}_k|\mathcal{A}_\nu)}{p(\mathcal{G}_k|\mathcal{A}_\nu) + p(\mathcal{G}_k|\neg\mathcal{A}_\nu)}. \quad (14)$$

By interactive learning, user-specific semantic posterior probabilities $p(\mathcal{A}_\nu|\mathcal{G}_k)$ are obtained for each subgraph $\mathcal{G}_k$. Therefore, a semantic labeling of subgraphs is carried out which enables spatio-temporal reasoning and probabilistic retrieval of spatio-temporal structures in SITS.

Fig. 9. Retrieval of Colza crops: most likely retrieved spatio-temporal structures ranked according to their posterior probabilities. Retrieved spatial masks (left) are associated to identical time-periods (the full SITS). Thus, the images displayed were selected from SITS in order to represent significantly the spatio-temporal structure behaviors.

V. Spatio-Temporal Reasoning

The hierarchical modeling concept for spatio-temporal reasoning has been applied on the ADAM dataset described in Section I-A. In the experiments carried out, the features were the three spectral reflectances extracted out of a spatial subset of 200×200 pixels. In the following, we show examples of query results obtained by the interactive training of several semantics.

We start with the analysis of ploughing semantics. We performed this training using examples of spatio-temporal structures defined in a time window of four samples. Fig. 7

presents the retrieved spatio-temporal structures of the highest semantic posterior probabilities together with their representations in SITS. Almost all other ploughing phenomena were successively retrieved. However, as ploughings can differ slightly according to the crop nature, and as there were only few examples of such events in the considered spatial window, the learning induced low posterior probabilities.

The training of a maturation semantics, specific to a field, was performed. As this phenomenon occurred over a long time period, a time window of 12 samples was selected for training. Contrary to the ploughland semantics, the interactive learning process, with very few positive and negative examples, enabled the retrieval of similar events with high posterior probabilities. The retrieved spatio-temporal structures are presented in Fig. 8 together with three significant image time samples. Note that, the crop evolutions of highest probabilities are maturation phenomena corresponding to the specific sought culture, whereas retrieved events with lower probabilities correspond to maturation of similar but slightly different cultures.

A last training was performed for the retrieval of Colza crops. We selected a time-window of 38 time samples, that is to say the longest possible time-period. Results are displayed in Fig. 9. Two Colza crops with very high probabilities were obtained by this search. The other retrieved structures were similar but the cultures were slightly different and were associated to lower probabilities.

VI. Conclusion

In this paper, we have presented generic methods for modeling hierarchically high-resolution SITS information content. The developed algorithms have been integrated in a novel tool dedicated to the exploration of SITS archives.

The concept relies on the unsupervised learning of dynamic cluster trajectories and on the interactive learning of users-specific interest. The modeling of trajectories in the dynamic feature space results in graphs coding synthetically spatio-temporal structures contained in SITS. To infer the graphs, Bayesian methods and entropic measurements have been applied on strong families of stochastic models. In the last years, information theoretical methods showed that entropy measures enable inference preserving the relevance of the information in the models. The relationship of these coding methods with Bayesian inference [10], [18] demonstrate the relevance of the model selection and parameter estimation approach. Based on the objective graph representation, the user-specific interest is interactively learned by positive and negative examples. The interactive learning procedure results in the semantic labeling of SITS and enables the probabilistic retrieval of similar spatio-temporal structures.

The hierarchical modeling procedure has been applied on a SPOT image time-series. The interactive learning of several semantics has demonstrated the relevance of the methods. Based on these experiments, the algorithms appears to be fast and relevant for the retrieval of user-specific spatio-temporal patterns.

Appendix I
Two-Part MDL Code Length for Gaussian Mixture Models

The two-part MDL code for Gaussian mixture models comprises the following description lengths, calculated using Shannon coding theorem.

- Let L_0 denote the code length needed to encode the number K of Gaussians of the mixture, within an interval of integers $[1, K_{\max}]$. Using the Shannon coding theorem with a uniform prior, we define this code length as
$$L_0 = \log_2(K_{\max}).$$
- Let L_1 denote the code length needed to encode the number of points $\{n_k, k \in [1, K]\}$ associated with the different Gaussians $\{G_k, k \in [1, K]\}$. Employing a uniform prior on the interval of integers $n_k \in [1, n]$, where n denotes the sample size, we obtain
$$L_1 = K \log_2(n).$$
- For each Gaussian G_k, the number of bits needed to encode its estimated mean vector and covariance matrix is calculated using a uniform probability distribution. In case of continuous distribution, to encode each parameter of the l-dimensional Gaussian G_k, we need a precision $\delta_k = 1/\sqrt{n_k}$ [10]. Considering the expectation of the number of points $n_k = \pi_k * n$ associated with the Gaussian G_k, in order to encode all the Gaussian parameters we need a number of bits equal to
$$L_2 = \sum_{k=1}^{K} \frac{l(l+1)}{2} \log_2(n\pi_k).$$
- The code length needed for the encoding of the n data point assignments to their respective clusters is given by
$$L_3 = \sum_{k=1}^{K} -n\pi_k \log_2 \pi_k.$$
- The second part composing the two-part MDL code is the number of bits needed to encode the data points $\boldsymbol{\psi} = \{\psi^1, \ldots, \psi^n\}$ knowing the complexity and the parameters of the mixture. It is given by
$$L_4 = \sum_{k=1}^{K} \sum_{\psi_i \in G_k} \left(-\frac{1}{2} \log_2 \frac{|A_k^{-1}|}{2\pi^l} + \frac{1}{ln(2)} {}^t(\psi^i - M_k) A_k^{-1} (\psi^i - M_k) \right).$$

Therefore, adding the two parts, we obtain a two-part MDL code for the Gaussian mixture model defined by (15), shown at the bottom of the next page.

Appendix II
Change of the Two-Part MDL Code Length for Gaussian Mixture Modeling Optimization

The notations of Section III-B2 and of Appendix I will be used in the following. According to (9), the two-part MDL code,

encoding $\boldsymbol{\psi}$ with a Gaussian mixture model, is defined by the expression

$$\begin{aligned} L_{2P}(\boldsymbol{\psi}) &= \log_2(K_{\max}) + K\log_2(n) \\ &\quad + \sum_{k=1}^{K} \frac{l(l+1)}{2}\log_2(n\pi_k) + \sum_{k=1}^{K} -n\pi_k \log_2 \pi_k \\ &\quad + \sum_{k=1}^{K} \sum_{\psi_i \in G_k} \log_2\left(\widehat{p}_l(\psi_i)\right) \\ &= L_0 + L_1 + L_2 + L_3 + L_4. \end{aligned} \tag{16}$$

Let $\Delta_{G_{k'}}(L_1), \Delta_{G_{k'}}(L_2), \Delta_{G_{k'}}(L_3)$, and $\Delta_{G_{k'}}(L_4)$ denote the code length change of L_1, L_2, L_3, and L_4 induced by the removal of a given Gaussian $G_{k'}$ in the mixture. Note that L_0 is constant. Therefore,

$$\begin{aligned} \Delta_{G_{k'}}\left(L_{2P}(\boldsymbol{\psi})\right) &= \Delta_{G_{k'}}(L_1) + \Delta_{G_{k'}}(L_2) \\ &\quad + \Delta_{G_{k'}}(L_3) + \Delta_{G_{k'}}(L_4). \end{aligned} \tag{17}$$

$\Delta_{G_{k'}}(L_1)$ is equal to the code length decrease obtained by saving the encoding of the number of points $n_{k'}$

$$\Delta_{G_{k'}}(L_1) = -\log_2(n). \tag{18}$$

$\Delta_{G_{k'}}(L_2)$ represents the code length decrease induced by saving the Gaussian parameter encoding

$$\Delta_{G_{k'}}(L_2) = -\frac{l(l+1)}{2}\log_2(n\pi_{k'}). \tag{19}$$

$\Delta_{G_{k'}}(L_3)$ represents the code length decrease induced by saving the new assignments of data belonging to the removed Gaussian. It is defined by $\Delta_{G_{k'}}(L_3)$ which is equal to the expression

$$-\sum_{j=1, j\neq k'}^{K^{(q)}} n(\pi_j + \pi_{jk'})\log_2(\pi_j + \pi_{jk'}) + \sum_{j=1}^{K^{(q)}} n\pi_j \log_2 \pi_j$$

where $\pi_{jk'}$ is the ML weight estimate of G_j, after removal of $G_{k'}$. Thus, this quantity is expressed by

$$\begin{aligned} \Delta_{G_{k'}}(L_3) &= n\pi_{k'}\log_2 \pi_{k'} - \sum_{j=1, j\neq k'}^{K^{(q)}} \\ &\quad \cdot n\left((\pi_j + \pi_{jk'})\log_2(\pi_j + \pi_{jk'}) - \pi_j \log_2 \pi_j\right). \end{aligned} \tag{20}$$

$\Delta_{G_{k'}}(L_4)$ represents the code length increase induced by encoding the assigned data with the new Gaussian mixture distribution. The reassignments of the data points which belonged to the removed Gaussian induce a change in the distribution of the Gaussian mixture. Thus, the code lengths related to the encoding of all of the data points according to their assigned new Gaussian distributions have to be reevaluated. Therefore,

$$\begin{aligned} \Delta_{G_{k'}}(L_4) &= \sum_{\psi_i \in G_{k'}} \log_2 \frac{p_{k'}(\psi_i)}{p_{k'k'}(\psi_i)} \\ &\quad + \sum_{j=1, j\neq k'}^{K^{(q)}} \sum_{\psi_i \in G_j} \log_2 \frac{p_j(\psi_i)}{p_{jk'}(\psi_i)} \end{aligned} \tag{21}$$

where $p_j(.)$ and $p_{jk'}(.)$ are respectively the ML estimated probability distribution of G_j before and after removal of $G_{k'}$. $p_{k'k'}(\psi_i)$ is the ML estimated distribution of the most probable Gaussian which is assigned to ψ_i after removal of $G_{k'}$. The first sum represents the code length change related to the encoding of the data points belonging to the removed Gaussian while the second term represents the code length change of the other data points induced by the change of the Gaussian mixture distribution. Because of an exaggerated computational cost, the ML estimated Gaussian mixture distribution after removal of the Gaussian component $G_{k'}$ is estimated by performing a single iteration of an EM algorithm.

The global code length change is obtained using (17), that is to say by adding these four terms. The resulting sum is given in (10).

Appendix III
Mutual Information Analytical Calculation

The change of MT clusters between consecutive times can be quantized by mutual information measurements. The mutual information between two continuous random variables X and Y is defined as

$$MI(X,Y) = \int p(x,y)\log\frac{p(x,y)}{p(x)p(y)}dxdy. \tag{22}$$

According to previous notations, MT cluster C^k_{MT} projections in consecutive times t_i, t_{i+1} are denoted by $C^k_{MT_{t_i}}$ and $C^k_{MT_{t_{i+1}}}$. These projections are multidimensional Gaussians which are marginal distributions related to an MT cluster. The joint distribution, denoted by $C^k_{MT_{t_i \cup t_{i+1}}}$, is obtained by projecting the MT cluster in a space composed of the two TL feature components. Thus, as the marginal and joint distributions are Gaussian, mutual information can be calculated analytically as

$$MI\left(C^k_{MT_{t_i}}, C^k_{MT_{t_{i+1}}}\right) = \frac{1}{2}\log_2 \frac{\left|A^k_{t_i}\right|\left|A^k_{t_{i+1}}\right|}{\left|A^k_{t_i \cup t_{i+1}}\right|} \tag{23}$$

$$\begin{aligned} L_{2P}(\boldsymbol{\psi}) &= \log_2(K_{\max}) + K\log_2(n) + \sum_{k=1}^{K}\left(\frac{l(l+1)}{2}\log_2(n\pi_k) - n\pi_k\log_2\pi_k\right. \\ &\quad \left. + \sum_{\psi_i \in G_k}\left(\frac{1}{2}\log_2\frac{\left|A_k^{-1}\right|}{2\pi^l} - \frac{1}{ln(2)}{}^t(\psi^i - M_k)A_k^{-1}(\psi^i - M_k)\right)\right) \end{aligned} \tag{15}$$

where $|A^k_{t_i}|$, $|A^k_{t_{i+1}}|$, and $|A^k_{t_i t_{i+1}}|$ denote the covariance matrix determinants attached to the MT cluster Gaussian projections $C^k_{MT_{t_i}}$, $C^k_{MT_{t_{i+1}}}$, and $C^k_{MT_{t_i \cup t_{i+1}}}$ [15].

ACKNOWLEDGMENT

The authors would like to thank A. Giros for stimulating discussions and for carefully preprocessing the data. The data were made available by the French Space Agency *(CNES)* from the ADAM project.

REFERENCES

[1] C. M. Antunes, "Temporal data mining: An overview," presented at the *Workshop on Temporal Data Mining With the Int. Conf. Knowledge Discovery and Data Mining*, 2001.

[2] P. Tan and C. Potter, "Finding spatio-temporal patterns in earth science data," presented at the *Workshop on Temporal Data Mining With the Int. Conf. Knowledge Discovery and Data Mining*, 2001.

[3] M. Schroeder, H. Rehrauer, K. Seidel, and M. Datcu, "Interactive learning and probabilistic retrieval in remote sensing image archives," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 5, pp. 2288–2298, Sep. 2000.

[4] R. Duda, P. Hall, and D. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2000.

[5] J. P. Benzecri *et al.*, *L'Analyse des Données (tomes 1 et 2)*. Paris, France: Paris Dunod, 1973.

[6] T.-W. Lee, M. Girolami, and T. J. Sejnowski, "A unifying framework for independent component analysis," *Comput. Math. Appl.*, vol. 39, no. 11, pp. 1–21, Mar. 2000.

[7] P. J. Huber, "Projection prsuit (with discussion)," *Ann. Statist.*, vol. 13, no. 2, pp. 435–475, 1985.

[8] C. Posse, "An effective two-dimensional projection pursuit algorithm," *Comm. Statist. Simul. Comput.*, vol. 19, pp. 1143–1164, 1990.

[9] J. H. Friedman, "Exploratory projection pursuit," *J. Amer. Statist. Assoc.*, vol. 82, pp. 249–266, 1987.

[10] J. Rissanen, "Stochastic complexity and modeling," *Ann. Statist.*, vol. 14, pp. 1080–1100, 1986.

[11] H. Bischof, A. Leonardis, and A. Selb, "MDL principle for robust vector quantization," *Pattern Anal. Appl.*, vol. 2, p. 5972, 1999.

[12] C. S. Wallace and D. L. Dowe, "Intrinsic classification by MML—the snob program," in *Proc. 7th Austr. Joint Conf. Artificial Intelligence*, 1994, pp. 37–44.

[13] P. Cheeseman, J. Kelly, M. Self, J. Stutz, W. Taylor, and D. Freeman, "AutoClass: a Bayesian classification system," in *Proc. 5th Int. Conf. Machine Learning*, Ann Arbor, MI, 1988, pp. 54–56.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *J. R. Statist. Soc.*, vol. B 39, pp. 1–38, 1977.

[15] S. Kullback, *Information Theory and Statistics*. New York: Dover, 1968.

[16] P. Heas and M. Datcu, "Bayesian learning on graphs for reasoning on image time-series," in *AIP Conf. Proc., Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, R. Fisher, R. Preuss, and U. von Tousaint, Eds. Melville, NY: Amer. Inst. Phys., 2004, vol. 735, pp. 127–137.

[17] H. Bunke and G. Allerman, "Inexact graph matching for structural pattern recognition," *Pattern Recognit. Lett.*, vol. 1, no. 4, pp. 245–253, 1983.

[18] P. Vitanyi and M. Li, "Minimum description length induction, Bayesianism, and Kolmogorov complexity," *IEEE Trans. Inf. Theory*, vol. 46, no. 2, pp. 446–464, Mar. 2000.

Patrick Héas received the master degree in computer science and telecommunication from the National Aerospace Engineering School (ENSAE), Toulouse, France, in 2001. He is currently pursuing the Ph.D. degree at the Institut de Recherche en Informatique de Toulouse (IRIT).

In 2001, he worked on the segmentation of astronomical images at the Space Center for Radiation Studies (CESR), Toulouse. From 2001 to 2003, he developed stochastic modeling algorithms with the Image Analysis Group at the German Aerospace Center (DLR), Oberpfaffenhofen. Since April 2003, he has been working with the Image Analysis and Interpretation Group at IRIT in collaboration with French Space Agency (CNES) on content-based retrieval and information mining in image time-series. His research interests are in Bayesian learning, information theory, spatio-temporal analysis, and dynamic scene understanding.

Mihai Datcu (SM'04) received the M.S. and Ph.D. degrees in electronics and telecommunications from the University "Politehnica" of Bucharest UPB, Bucharest, Romania, in 1978 and 1986, and the title "Habilitation a diriger des recherches" from Université Louis Pasteur, Strasbourg, France.

He holds a Professorship in electronics and telecommunications with UPB since 1981. Since 1993, he has been a Scientist with the German Aerospace Center (DLR), Oberpfaffenhofen, Germany, and is currently a Senior Scientist and Image Analysis Research Group Leader with the Remote Sensing Technology Institute IMF of DLR. He is currently developing algorithms for model-based information retrieval from high-complexity signals, methods for scene understanding from SAR and interferometric SAR data, and he is engaged in research in information theoretical aspects and semantic representations in advanced communication systems. He has held Visiting Professor appointments from 1991 to 1992 with the Department of Mathematics, University of Oviedo, Oviedo, Spain, from 2000 to 2002 with the Université Louis Pasteur, and the International Space University, Strasbourg, France. In 1994, he was a Guest Scientist with the Swiss Center for Scientific Computing (CSCS). Manno, Switzerland, and in 2003, he was a Visiting Professor with the University of Siegen, Siegen, Germany. From 1992 to 2002, he had longer Visiting Professor assignments with the Swiss Federal Institute of Technology (ETH), Zurich, Switzerland. He is involved in advanced research programs for information extraction, data mining and knowledge discovery, and data understanding with the European Space Agency, Centre National d'Etudes Spatiales, the National Aeronautics and Space Administration, and in a variety of European projects. He is currently a Senior Scientist and Image Analysis research group leader with the Remote Sensing Technology Institute (IMF), DLR. His research interests are in Bayesian inference, information and complexity theory, stochastic processes, model-based scene understanding, image information mining, for applications in information retrieval and understanding of high-resolution SAR and optical observations.