Transactions on Machine Learning
and Data Mining
Vol. 5, No. 1 (2012) 23-44

ISSN:1865-6781,
ISBN: 978-3-942952-11-8

ibai Publishing
www.ibai-publishing.org

Efficient Spatio-temporal Mining of Satellite Image Time Series for Agricultural Monitoring

Andreea Julea[1], Nicolas Méger[2], Christophe Rigotti[3], Emmanuel Trouvé[2], Romain Jolivet[4], and Philippe Bolon[2]

[1] Institute for Space Sciences, P.O. Box MG-23, Ro 077125, Bucharest-Măgurele, Romania
andreeamj@spacescience.ro
[2] Université de Savoie, Polytech Annecy-Chambéry, LISTIC Laboratory, BP 80439, F-74944 Annecy-le-Vieux Cedex, France
{nicolas.meger|emmanuel.trouve|philippe.bolon}@univ-savoie.fr
[3] Université de Lyon, CNRS, INRIA, INSA-Lyon, LIRIS, UMR 5205, F-69621, Lyon, France
christophe.rigotti@insa-lyon.fr
[4] Université Joseph Fourier, ISTerre Laboratory, CNRS - UMR 5559. B.P. 53, F-38041 Grenoble Cedex 09, France
romain.jolivet@ujf-grenoble.fr

Abstract. In this paper, we present a technique for helping experts in agricultural monitoring, by mining Satellite Image Time Series over cultivated areas. We use frequent sequential patterns extended to this spatio-temporal context in order to extract sets of connected pixels sharing a similar temporal evolution. We show that a pixel connectivity constraint can be partially pushed to prune the search space, in conjunction with a support threshold. Together with a simple maximality constraint, the method reveals meaningful patterns in real datasets.

Keywords: Satellite Image Time Series, Spatio-temporal Patterns, Constraints, Agricultural Monitoring

1 Introduction

Current environmental and economic problems require better large scale agricultural monitoring. The continuous development of acquisition techniques of

satellite images provides ever growing volumes of data containing precious information for environmental and agricultural remote sensing. It is now possible to gather series of images concerning a given geographical zone at a reasonable cost. This kind of datasets, termed as a Satellite Image Time Series (SITS), offers a great potential, but raises new analysis challenges as data volumes to be processed are large and noisy (e.g., atmospheric variations, presence of clouds), and as both the temporal and the spatial dimensions have to be taken into account.

We present an unsupervised technique to support SITS analysis in agricultural monitoring. This paper is an extension of [1]: examples are detailed and new experiments are presented. The presented approach relies on frequent sequential pattern extraction [2] along the temporal dimension, combined with a spatial connectivity criterion. It allows to uncover sets of pixels satisfying two properties of cultivated areas: they are spatially connected/grouped and share similar temporal evolutions. The approach requires no prior knowledge of the objects (identified regions) to monitor and needs no user-supplied aggregate functions nor distance definitions. It is based on the extraction of patterns, called *Grouped Frequent Sequential patterns (GFS-patterns)*, satisfying a support constraint and a pixel connectivity constraint.

In this paper, we extend the general framework of GFS-patterns we proposed in [3] in two directions, when applied to agricultural monitoring.

Firstly, we show that, even though the connectivity constraint does not belong to any typical constraint family (e.g., monotonic, anti-monotonic), it can be pushed partially in the search space exploration. This leads to significant reduction of execution times on real Satellite Image Time Series of cultivated areas.

Secondly, we show that a simple post-processing using a maximality constraint over the patterns is very effective. Indeed, it restricts the number of patterns to a human-browsable collection, while still retaining highly meaningful patterns for agro-modelling. This property is confirmed even for poor quality inputs (rough image quantization, raw noisy images).

The new extended approach seems particularly suitable in exploratory mining stages on this kind of data. Indeed, we show that, on optical SITS, the method can differentiate between cultivated fields and non-cultivated areas (city, path, field border), can find areas of homogeneous crops, and even highlight particular varieties of a crop or irrigation/fertilization differences. To our knowledge, no such coarse to fine grained results have been reported using a single other unsupervised method. The technique presented in this paper does not aim to be exhaustive (e.g., identifying groups for all crops or varieties), but requires no domain knowledge (except the use of the well-known *Normalized Difference Vegetation Index (NDVI)* [4]) and needs only a simple preprocessing of the SITS. Furthermore, we show that the approach is general enough to be applied, not only to optical SITS, but also to radar SITS to find ground deformation patterns that can be useful to plan long term soil usage.

Additionally, even though the building blocks of the approach are simple, their combination is effective, and could lead to other interesting applications,

and extension opportunities for the sequential pattern mining techniques developed during the past decade.

The rest of the paper is organized as follows. Related work is discussed in Section 2. GFS-patterns are introduced in Section 3. Their extraction and the partial push of the connectivity constraint are presented in Section 4. Qualitative experiments and performance on real optic data are reported in Section 5. Results on radar data are given in Section 6. We conclude with a summary in Section 7.

2 Related Work

SITS can be processed at a higher level than the pixel one, after having identified objects or groups of pixels forming regions of interest. For example, in [5], stochastic models such as Gibbs-Markov random fields are used to extract spatial and spectral features of objects/regions over time. Spatio-temporal patterns can also be extracted from SITS by mining a sequence of signatures as proposed in [6]. In this case, self-organizing maps are used to extract signatures of regions in each image, and then the SITS is encoded as a sequence of region signatures. This sequence is further searched under temporal and frequency constraints to find a kind of serial episode-based rules [7] such as "$A \Rightarrow B$" which can be read as "if signature A is observed once or more, then, sometimes later, signature B is observed once or more". This family of approaches, needs as input identified objects/regions (already identified). If not known, objects/regions are hard to select in SITS since groups of pixels do not always form objects in a single image[1] (e.g., because of atmospheric perturbations, shading phenomena).

Per-pixel SITS analysis techniques have also retained attention as they do not require prior object identification. These techniques are essentially clustering ones. The feature vector associated to each pixel, and used to compare them, can contain aggregated values over time (e.g., average or min/max of the values associated to the pixel) as in [8], if the user has some insight about the kind of aggregates that is appropriated. The feature vector can also be the whole vector of values associated to the pixel, leading to perform a clustering in a high dimensional space. Such a clustering can be difficult to interpret and requires a careful parameter setting [9] as well as sophisticated distances such as the adaptation of the Levenshtein edit distance proposed in [10] to measure the distance between the sequences of values associated to pixels. These approaches are the closest to the one presented in this paper, in the sense that they perform per-pixel analysis without prior knowledge of the objects (identified regions) to monitor. However, they required to incorporate domain knowledge in the form of feature/aggregation/distance definitions, and they do not find overlapping areas or areas that refine other areas, such as the ones that will be presented in Section 5.

[1] This cannot be easily overcome, for instance, by averaging pixel values over consecutive images, since the aspect of an object is likely to change from a image to the next one.

Other approaches, based on change detection, generate a single image in which changes are plotted, i.e., a change map. Change detection techniques generally require prior information about the type of changes that has to be taken into account, and are targeted to a specific phenomenon. For example, one may want to look for abrupt changes such as floods, earthquakes or anthropic disasters (e.g., [11]), while others may be interested in gradual changes such as biomass accumulation (e.g., [12]). Change detection techniques can be applied efficiently at the pixel level (e.g., [13], [14]), and can also be done at the texture level as proposed in [15] or at the object level (e.g., [16]).

Other works (e.g., [17–20]) also rely on local patterns to analyze spatio-temporal datasets. Nevertheless, to our knowledge, they reported no application to satellite image time series. In [17], frequent sequential patterns representing sub-trajectories of objects, i.e., sequences of spatial locations are mined. In [18], one single object is considered and its trajectory is represented as a large event sequence from which periodic sub-trajectories that are frequent enough are extracted. Trajectory mining can also be achieved using other patterns and measures. For example, in [21], a pattern is a group of objects sharing a common kind of motion (direction, speed) at a given date within a same space portion. Global models can also be used for mining trajectories as proposed in [20], where the authors present a density-based clustering algorithm for moving object trajectories. All these techniques could be applied to SITS to analyze trajectories, after having identified the objects of interest. In [19], frequent sequential patterns are used to represent spatio-temporal relations in data point neighborhoods. For example, if a sequential pattern "$A \rightarrow B$" is found, then it is interpreted as "data points of type B tend to occur around and after data points of type A". In its principle, with appropriated data encoding, this approach could also be adapted to SITS, leading to patterns that could describe pixel neighborhood evolutions, like shifts due to motions or area expansions.

3 Grouped Frequent Sequential Patterns

In this section, *grouped frequent sequential patterns* are introduced. They are dedicated to the extraction of groups of pixels, in which the pixels in a group share a common temporal pattern and satisfy a minimum average connectivity over space. Firstly, some preliminary definitions are given so as to view a SITS as a set of temporal sequences. Secondly, we recall and adapt to this context a common kind of local patterns, the so-called *sequential patterns*. Then, in the third part of this section, the connectivity measure used to define the grouped frequent sequential patterns is introduced.

3.1 Preliminary Definitions

Let us consider a SITS, i.e., a satellite image time series that covers the same area at different dates. Within each image, each pixel is associated with a value, e.g., the reflectance intensity of the geographical zone it represents. We transform

these pixel values into values belonging to a discrete domain, using labels to encode pixel states. These labels can correspond to ranges obtained by image quantization or to pixel classes resulting from an unsupervised classification (e.g., using K-means or EM-based clustering).

Definition 1. ***(label and pixel state)*** Let $L = \{i_1, i_2, \ldots, i_s\}$ be a set containing s distinct symbols termed *labels*, and used to encode the values associated to pixels. A *pixel state* is a pair (e, t) where $e \in L$ and $t \in \mathbb{N}$, and such that t is the occurrence date of e. Date t is simply the timestamp of the image from which the value e has been obtained.

Then, we define a *symbolic SITS* as a set of *pixel evolution sequences*, each sequence describing the states of a pixel over time.

Definition 2. ***(pixel evolution sequence and symbolic SITS)*** For a pixel p, the *pixel evolution sequence* of p is a pair $((x, y), seq)$, where (x, y) are the coordinates of p and seq is a tuple of pixel states $seq = \langle (e_1, t_1), (e_2, t_2), ..., (e_n, t_n) \rangle$ containing the states of p ordered by increasing dates of occurrences. A *symbolic SITS* (or SITS when clear from the context) is then a set of pixel evolution sequences.

For a typical symbolic SITS, we thus get a set of millions of pixel evolution sequences, each sequence containing the discrete descriptions of the values associated to a given pixel over the time. A toy symbolic SITS containing the states of four pixels is given by Example 1.

Example 1

$$\begin{gathered}((0,0), \langle (1,A), (2,B), (3,C), (4,B), (5,D) \rangle), \\ ((0,1), \langle (1,B), (2,A), (3,C), (4,B), (5,B) \rangle), \\ ((1,0), \langle (1,D), (2,B), (3,C), (4,B), (5,C) \rangle), \\ ((1,1), \langle (1,C), (2,A), (3,C), (4,B), (5,A) \rangle)\end{gathered}$$

This dataset describes the evolution of four pixels located at positions $(0,0)$, $(0,1)$, $(1,0)$ and $(1,1)$. Dates $1, 2, 3, 4$ and 5 are considered: five successive images are taken into account. Pixel states are described using symbols A, B, C and D. For example, the successive pixel states of the pixel located at $(1,0)$ are D, B, C, B and C.

3.2 Sequential Patterns

A typical base of sequences is a set of sequences of discrete events, in which each sequence has a unique sequence identifier. Regarding SITS, if we take the pairs (x,y) of coordinates of pixels as identifiers of their evolution sequences, then a

symbolic SITS is a base of sequences, and the standard notions [2] of sequential patterns and sequential pattern occurrences can be easily defined as follows[2].

Definition 3. ***(sequential pattern)*** A *sequential pattern* α is a tuple $\langle \alpha_1, \alpha_2, \ldots, \alpha_m \rangle$ where $\alpha_1, \ldots, \alpha_m$ are labels in L and m is the *length* of α. Such a pattern is also denoted as $\alpha_1 \rightarrow \alpha_2 \rightarrow \ldots \rightarrow \alpha_m$.

Definition 4. ***(occurrence and support)*** Let $\mathcal{S}$ be a symbolic SITS, and $\alpha = \alpha_1 \rightarrow \alpha_2 \rightarrow \ldots \rightarrow \alpha_m$ be a sequential pattern. Then $((x, y), \langle (\alpha_1, t_1), (\alpha_2, t_2), \ldots, (\alpha_m, t_m) \rangle)$, where $t_1 < t_2 < \ldots < t_m$, is an *occurrence* of α in $\mathcal{S}$ if there exists $((x, y), seq) \in \mathcal{S}$ such that (α_i, t_i) appears in seq for all i in $\{1, \ldots, m\}$. Such a pixel evolution sequence $((x, y), seq)$ is said to support α. The *support* of α in $\mathcal{S}$, denoted by $support(\alpha)$, is the number of sequences in $\mathcal{S}$ that support α.

When considering Example 1, the occurrences of sequential pattern $A \rightarrow C \rightarrow B$ are (notice that the elements in an occurrence do not need to be contiguous in time):

$$((0,0), \langle (1, A), (3, C), (4, B) \rangle),$$
$$((0,1), \langle (2, A), (3, C), (4, B) \rangle),$$
$$((0,1), \langle (2, A), (3, C), (5, B) \rangle),$$
$$((1,1), \langle (2, A), (3, C), (4, B) \rangle)$$

Indeed, pattern $A \rightarrow C \rightarrow B$ occurs in the pixel evolution sequence of the pixel located at position $(0, 1)$. Label A occurs at date 2, label C occurs at date 3 and label B occurs at date 4 and date 5: two different occurrences can thus be considered for the pixel located at position $(0, 1)$. This is not the case for all other pixels. More precisely, pattern $A \rightarrow C \rightarrow B$ occurs only once for pixels located at position $(0, 0)$ and $(1, 1)$, and no occurrence is observed for the pixel located at position $(1, 0)$. Though four occurrences can be found in the dataset of Example 1, pattern $A \rightarrow C \rightarrow B$ only appears in three different pixel evolution sequences. In other words, pattern $A \rightarrow C \rightarrow B$ affects only three different pixels. Thus, its support is $support(A \rightarrow C \rightarrow B) = 3$. Finally, it should be pointed out that a label can be repeated within a pattern, and for instance, pattern $C \rightarrow C$ has two occurrences, one in the third (pixel position $(1, 0)$) and one in the fourth (pixel position $(1, 1)$) sequence.

Definition 5. ***(frequent sequential pattern)*** Let σ be a strictly positive integer termed a *support threshold*. Let α be a sequential pattern, then α is a *frequent sequential pattern* if $support(\alpha) \geq \sigma$. The support threshold can also be specified as a relative threshold $\sigma_{rel} \in [0, 1]$. Then a pattern α is frequent if $support(\alpha)/|\mathcal{S}| \geq \sigma_{rel}$, where $\mathcal{S}$ is the dataset and $|\mathcal{S}|$ is the number of sequences in $\mathcal{S}$.

[2] Notice that, in the original definitions, several elements can occur at the same time in a sequence, while in our context a timestamp is associated to a single element.

Reusing the definitions of sequential patterns and of sequential patterns occurrences helps us to take advantage of the great research effort made in this domain to develop efficient extraction techniques (e.g., [2, 22–28]).

3.3 Spatial Connectivity

The way sequential patterns are applied to SITS analysis leads to a natural interpretation of the notion of support. In fact, for a pattern α, the support of α is simply an area, i.e., the total number of pixels in the image having an evolution in which α occurs. These pixels are said to be *covered* by α.

Definition 6. ***(covered pixel)*** A pixel associated to the evolution sequence $((x, y), seq)$ is *covered* by a sequential pattern α if α has at least one occurrence in seq. The set of the coordinates of the pixels covered by α is denoted by $cover(\alpha)$. By definition, $|cover(\alpha)| = support(\alpha)$.

However, a threshold on the covered area is not sufficient, because, most of the time, interesting parts in images are made of pixels forming regions in space. Thus, an additional criterion, the *average connectivity* measure, based on the *8-nearest neighbors (8-NN)* convention [29], is introduced. This measure enables to select patterns that cover pixels having a tendency to form groups in space. It is defined as follows:

Definition 7. ***(local connectivity)*** For a symbolic SITS $\mathcal{S}$, let $occ((x, y), \alpha)$ be a function that, given the spatial coordinates (x, y) and a sequential pattern α, indicates whether α occurs in $\mathcal{S}$ at location (x, y). More precisely, $occ((x, y), \alpha)$ is equal to 1 if and only if there is a sequence seq in $\mathcal{S}$ at coordinates (x, y) and α occurs in $((x, y), seq)$. Otherwise $occ((x, y), \alpha)$ is equal to 0. If α occurs in $((x, y), seq)$, then its *local connectivity* at location (x, y) is $LC((x, y), \alpha) = [\sum_{i=-1}^{i=1} \sum_{j=-1}^{j=1} occ((x + i, y + j), \alpha)] - 1$.

The value $LC((x, y), \alpha)$ is the number of pixels in the 8-neighborhood of (x, y) that have an evolution supporting α. The reader should notice that the sum is decremented by one, so as not to count the occurrence of α at location (x, y) it-self. In Example 1, for sequential patterns $A \to C \to B$ and $C \to C$ we have:

$$\begin{gathered} LC((0,0), A \to C \to B) = 2 \\ LC((0,1), A \to C \to B) = 2 \\ LC((1,1), A \to C \to B) = 2 \\ LC((0,1), C \to C) = 1 \\ LC((1,1), C \to C) = 1 \end{gathered}$$

Definition 8. ***(average connectivity)*** The *average connectivity* of α is defined as:

$$AC(\alpha) = \frac{\sum_{(x,y)\in cover(\alpha)} LC((x,y),\alpha)}{|cover(\alpha)|}$$

This measure gives, for the pixels supporting α, the average number of neighbors in their 8-NN that also support α. In Example 1, $AC(A \to C \to B) = 6/3 = 2$ and $AC(C \to C) = 2/2 = 1$. Finally, we define the *grouped frequent sequential patterns* as follows.

Definition 9. ***(GFS-pattern)*** Let $\mathcal{S}$ be a symbolic SITS, given a sequential pattern α frequent in $\mathcal{S}$, and a positive real number κ termed *average connectivity threshold*, α is said to be a *Grouped Frequent Sequential pattern* (*GFS-pattern*) if $AC(\alpha) \geq \kappa$ in $\mathcal{S}$.

For instance, in Example 1, if $\sigma = 2$ and if $\kappa = 2$, then $A \to C \to B$ is a grouped frequent sequential pattern while $C \to C$ is not.

4 Grouped Frequent Sequential Pattern Extraction

As mentioned in Section 3, several efficient techniques for extracting sequential patterns in a base of sequences are available and can be used in our context. A naive solution is to extract frequent sequential patterns and then, in a post-processing step, to select among them the ones satisfying the average connectivity constraint $AC(\alpha) \geq \kappa$. In this section, we show that this constraint can be pushed partially in the extraction process to prune the search space and reduce extraction times, as reported in the experiment presented in Section 5.2.

The average connectivity constraint does not correspond to a class of constraints that has been identified in sequential pattern mining, and for which pruning techniques have been proposed. The two main classes of constraints are the anti-monotonic constraints (if a pattern does not satisfy the constraint then its super-patterns cannot satisfy it) and monotonic constraints (if a pattern satisfies the constraint then all its super-patterns satisfy it).

For the simple form of sequential patterns used in this paper, the notion of *super-patterns* can be defined as follows.

Definition 10. ***(super-pattern)*** A sequential pattern $\beta = \beta_1 \to \beta_2 \to \ldots \to \beta_m$ is a *super-pattern* of a sequential pattern $\alpha = \alpha_1 \to \alpha_2 \to \ldots \to \alpha_n$ if there exist integers $1 \leq i_1 < i_2 < \ldots < i_n \leq m$ such that $\alpha_1 = \beta_{i_1}$, $\alpha_2 = \beta_{i_2}$, ..., $\alpha_n = \beta_{i_n}$.

It is straightforward that the average connectivity constraint is neither anti-monotonic, nor monotonic, and it is easy to show that it is neither prefix anti-monotonic, nor prefix monotonic [25]. Moreover it does not belong to classes of constraints used for frequent pattern mining in general, such as succinct [30], convertible [31] or loose anti-monotone [32].

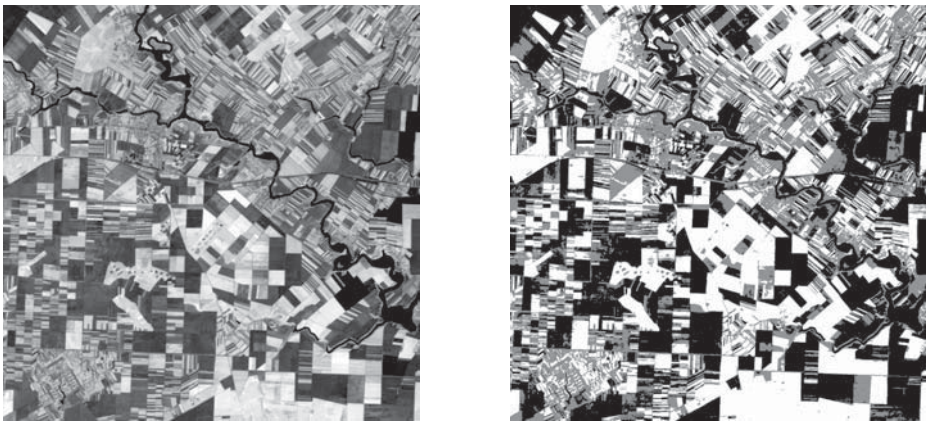
The key hints to push partially the average connectivity constraint is to observe that for any frequent sequential pattern α since $|cover(\alpha)| \geq \sigma$, then

$$AC(\alpha) = \frac{\sum_{(x,y)\in cover(\alpha)} LC((x,y),\alpha)}{|cover(\alpha)|} \leq \frac{\sum_{(x,y)\in cover(\alpha)} LC((x,y),\alpha)}{\sigma}$$

Thus a frequent pattern α that does not satisfy $\frac{\sum_{(x,y)\in cover(\alpha)} LC((x,y),\alpha)}{\sigma} \geq \kappa$ cannot be a GFS-pattern. And, if we consider the conjunction of constraints $\mathcal{C} = support(\alpha) \geq \sigma \wedge \frac{\sum_{(x,y)\in cover(\alpha)} LC((x,y),\alpha)}{\sigma} \geq \kappa$, this conjunction is anti-monotonic, since the value $\sum_{(x,y)\in cover(\alpha)} LC((x,y),\alpha)$ cannot increase for super-patterns of α, and thus this conjunction can be used actively to prune the search space.

There is no real need for a new extraction algorithm, since many, if not all, of the sequential pattern mining algorithms can handle and push in the extraction process anti-monotonic constraints. We decided to integrate the anti-monotonic conjunction $\mathcal{C}$ into the *PrefixGrowth* algorithm [25], that is a recent and efficient algorithm for sequential pattern mining under constraints, and that can easily handle anti-monotonic constraints among others. Beside checking $\mathcal{C}$ to prune the search space, the only modification required is to verify, before outputting a pattern α, that $AC(\alpha) \geq \kappa$, since satisfying $\mathcal{C}$ does not imply satisfying the average connectivity constraint. The implementation of the whole algorithm has been done in C using our own data structures.

5 Experiments



a) b)

Fig. 1. Satellite images of Fundulea, Romania - a) NDVI image b) the same image after a 3-level quantization of NDVI values, using the 33rd and the 66th centiles.

We report experiments on the ADAM (Data Assimilation by Agro-Modeling) SITS [33], a SITS dedicated to the assessment of spatial data assimilation tech-

niques within agronomic models. This dataset and its preprocessing are presented in Section 5.1. The resulting dataset is a set of one million of sequences of size 20. In Section 5.2, we show that pushing the average connectivity measure constraint, during GFS-pattern extraction, is effective to reduce the search space. Then, in Section 5.3, we show that together with a maximality constraint, the approach is useful to find meaningful patterns in real data. All experiments have been run on a standard PC (Intel Core 2 @3GHz, 4 GB RAM, Linux kernel 2.6), using our own extractor engine developed in C (see Section 4).

5.1 The ADAM SITS: Selection and Preprocessing

We select a dataset of 20 images of the ADAM SITS taken between October 2000 and July 2001, so as to make sure that enough data is available to observe agricultural cycles, from autumn ploughing and seeding to harvest. These images have been acquired within three bands by SPOT satellites: B1 in green (500 - 590 nm), B2 in red (610 - 680 nm) and B3 in near infrared (NIR 780 - 890 nm). The spatial resolution is 20m×20m (per pixel) and the observed scene is a rural area located in East Bucharest, Romania. We choose a sub-scene of 1000×1000 pixels, depicting an area called Fundulea. The main interest in selecting this sub-scene, and this time period, is that the corresponding ground truth is available in this area over 2000-2001 for the fields belonging to the Romanian National Agricultural Research and Development Institute. These fields represent 5.9% of the scene and can be used to evaluate our results.

The dataset corresponding to this sub-scene contains noise (mainly atmospheric perturbations), and has a size that is typical in the domain of per-pixel SITS analysis (20 images of 1000×1000 pixels). The sub-scene mainly shows agricultural fields whose dimensions are larger than the spatial resolution. Various types of crops such as wheat, corn, barley, chickpea, soya, sunflower, pea, millet, or oats are present. Other objects can be categorized into 'roads', 'rivers', 'forests' and 'towns'. The topography of this region is generally flat with a very limited fraction of the area corresponding to slopes bordering a river and to several micro-depressions.

For each pixel, and for each date, we compute a synthetic band B4 corresponding to the *Normalized Difference Vegetation Index* (NDVI) [4] and defined as $B4 = \frac{B3-B2}{B3+B2}$. The NDVI index is widely used to detect live green plant canopies in multispectral remote sensing data. An example of an original image of the ADAM SITS encoded in the B4 band is presented in Figure 1a. The image quantization is performed by splitting the B4 value domain into 3 intervals using the 33rd and the 66th centiles. In order to minimize the influence of possible calibration defaults, quantization is separately done for each image. For a given acquisition date, a pixel is described by a single label that indicates which interval this pixel value belongs to. Label '1' relates to low NDVI values, label '2' represents mid NDVI values and label '3' denotes high NDVI values. The result of the quantization of the image of Figure 1a is shown in Figure 1b. When

encoded as sequences, we obtain a set of one million of sequences of size 20 over an alphabet of 3 symbols.

5.2 Quantitative Results

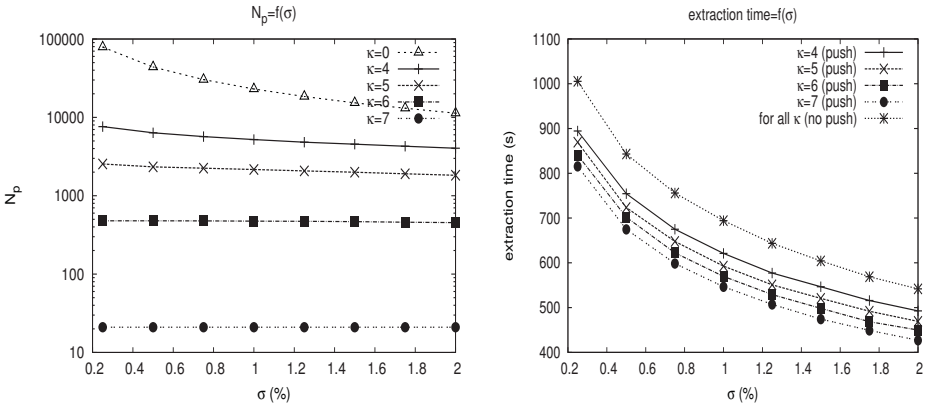
The two parameters that can be set by the user are σ, the minimum support and κ, the minimum average connectivity. The values of the minimum support are taken in the range $[0.25\%, 2\%]$ so as to ask for minimum areas covering from 2500 pixels (1 km^2) to 20000 pixels (8 km^2). These values allow us either to consider all fields including the smallest ones (low σ values) or to extract quite large fields (high σ values). In order to assess the impact of κ, values between 0 and 7 are considered. As the definition of the average connectivity measure relies on the 8-nearest neighbors convention, and makes no distinction between pixels on image borders and the other ones, the average connectivity measure indeed belongs to $[0, 8)$.

The experiments show that the number of frequent sequential patterns that are discarded thanks to the minimum average connectivity constraint is important, and that pushing partially this constraint leads to a significant reduction of the execution times (from 10 to 20%).

The number of output patterns N_p can be several orders of magnitude smaller than the total number of frequent patterns. This is represented in Figure 2a. If no minimum average connectivity constraint is applied ($\kappa = 0$), then all frequent sequential patterns are extracted, and N_p rises up to 78885 patterns, and as expected, the higher κ is, the lower is N_p. In the worst case scenario, i.e., for $\sigma = 0.25\%$, if $\kappa = 4$, then $N_p = 7623$ while if $\kappa = 7$ then $N_p = 21$. The minimum average connectivity constraint is a very selective one, as it can be observed, for a given value of κ such that $\kappa \neq 0$, N_p has rather limited variations with respect to σ. For example, for $\kappa = 4$, N_p rises from 4042 ($\sigma = 2\%$) to 7623 GFS-patterns ($\sigma = 0.25\%$) while for $\kappa = 6$, N_p rises from 454 ($\sigma = 2\%$) to 479 GFS-patterns ($\sigma = 0.25\%$). N_p is even stable for $\kappa = 7$ with 21 GFS-patterns.

As presented in Figure 2b, extraction times are the same for all values of κ if the average connectivity constraint is not pushed (one single curve). If the constraint is pushed, then extraction times are reduced for all settings, from 10% up to 20%. For example, for $\sigma = 0.75\%$ and $\kappa = 7$, it takes 756 seconds to perform an extraction without constraint pushing while it only takes 599 seconds with constraint pushing.

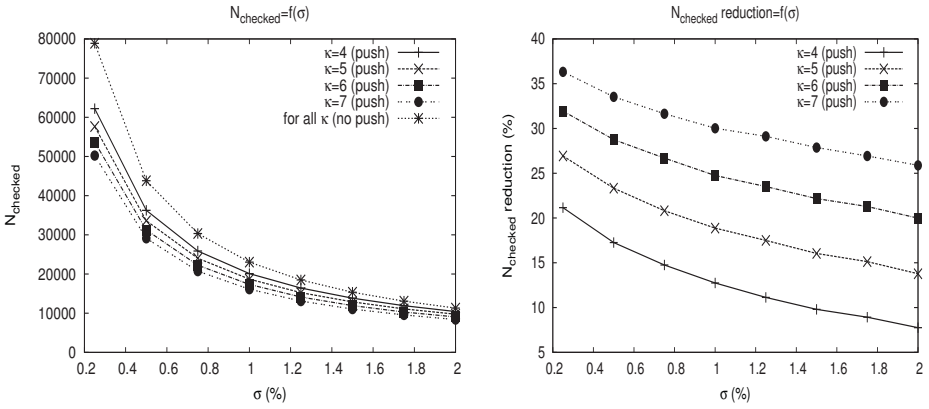
The corresponding pruning can be quantified using the number $N_{checked}$ of frequent sequential patterns that are considered during the extraction and for which the average connectivity constraint is checked. The values obtained for $N_{checked}$ are given Figure 3a. If no constraint pushing is performed, then, for example, $N_{checked}$ rises up to 78885 patterns for $\sigma = 0.25$ (whatever κ might be). At the same support threshold, if the constraint pushing is performed, then, for instance, with $\kappa = 7$, $N_{checked}$ goes down to 50227. For a given σ and a given κ, when the constraint is pushed, $N_{checked}$ is reduced in all settings. This reduction (in %) is depicted in Figure 3b. It varies between 7.7% ($\sigma = 2\%$,$\kappa = 4$)



a) b)

Fig. 2. Evaluation of the impact of the average connectivity constraint: a) N_p, the number of extracted patterns and b) extraction times (with and without constraint pushing) vs. κ, the minimum average connectivity threshold and σ, the minimum support threshold.

and 36.3% ($\sigma = 0.25\%$,$\kappa = 7$). The pruning is more effective (large relative reduction) in the most difficult extraction settings (low values of σ).



a) b)

Fig. 3. Evaluation of the constraint pushing: a) $N_{checked}$, the number of checked patterns and b) the corresponding $N_{checked}$ reduction vs. κ, the minimum average connectivity threshold and σ, the minimum support threshold.

5.3 Qualitative Results

In this section, σ is set to 1% in order to ask for GFS-patterns relating to areas covering at least 4 km^2 (the whole image covers 400 km^2). Main crops are thus focused on, which will help us in characterizing our results. The ground truth that has been made available by the experts and that covers 5.9% of the image indeed contains representative crops of that region.

We show that using a typical maximality constraint on these patterns is a very effective way to focus on a small number of meaningful GFS-patterns, still carrying key information for agro-modelling experts.

The maximality constraint used is very simple, it consists of selecting the patterns in the output having no super-pattern also present in the output. These patterns are in some sense the most specific ones.

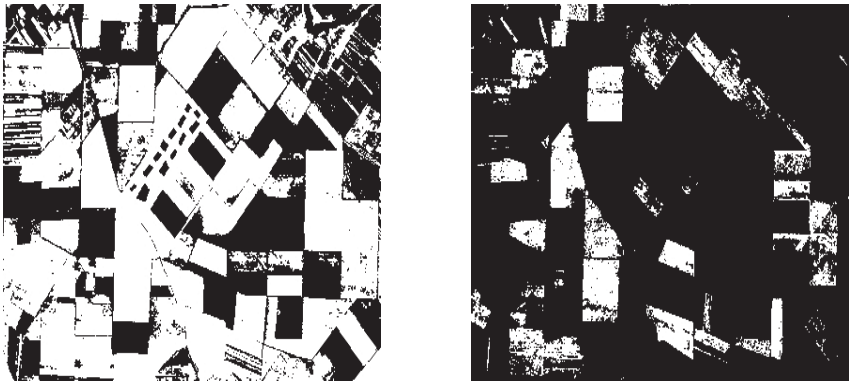
To visualize the result, for each of these maximal patterns, we draw an image where the pixels covered by the pattern are highlighted. Since we obtain only a few tens of such images, the visual inspection can be quickly done by the expert. Notice that if we extract all frequent sequential patterns (without taking into account the spatial connectivity of the pixels) at $\sigma = 1\%$, then 23038 patterns are obtained, among which 4684 are maximal, forming a collection that cannot be handled by the expert.

It should also be pointed out, that in these experiments, the image quantization does not seem to be a critical issue, as well as the presence of intrinsic noise in SITS (mainly atmospheric variations and clouds). Indeed, though the image quantization in 3 levels leads to patterns built over a small alphabet of 3 labels, and though no dedicated noise preprocessing is performed, the joint use of the spatial and temporal information still allows to find meaningful patterns. So the technique is likely to be applicable to poor quality image series (e.g., due to the limitations of the measuring devices) and to require little preprocessing.

For the first experiment we set κ to 7, which is a very selective value of the threshold. In this case, 21 GFS-patterns are obtained. They relate to general evolutions as their length does not exceed 12. Only 7 are maximal, and among them we have for example, pattern $3 \to 3 \to 3 \to 3 \to 3 \to 3$. The pixels covered by that pattern are depicted in white in Figure 4a over the area for which the ground truth is available. It covers 96.2% of the pixels of the ground truth that correspond to cultivated fields, and 98.3% of the pixels it covers in this area correspond to cultivated fields.

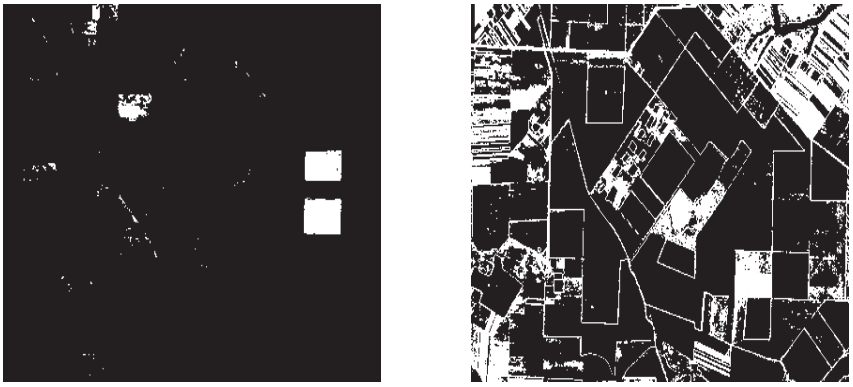
In order to get more specific evolutions, i.e., longer patterns, we set κ to a less selective value and use $\kappa = 6$. We obtain 31 maximal patterns out of the 474 GFS-patterns that are extracted. One of these maximal patterns is $2 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 1 \to 1 \to 1 \to 1$. The pixels covered by that pattern are represented in Figure 4b. According to the ground truth, it covers 61.4% of the pixels of the ground truth that relate to wheat crop, and 96.3% of the pixels it covers in the area where the ground truth is available, correspond to wheat crops.

Interesting information can be drawn from such patterns. For instance, as it



a) b)

Fig. 4. a) Localization (white pixels) of pattern/pixel evolution $3 \to 3 \to 3 \to 3 \to 3 \to 3$ (cultivated fields) b) Localization (white pixels) of pattern/pixel evolution $2 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 1 \to 1 \to 1 \to 1$ (wheat).



a) b)

Fig. 5. a) Localization (white pixels) of pattern/pixel evolution $3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 1 \to 1 \to 1 \to 1$ (particular variety of wheat) b) Localization (white pixels) of pattern/pixel evolution $2 \to 2 \to 2 \to 2 \to 2 \to 2 \to 2 \to 2$ (paths, fallows, cities and field borders).

can be observed, some *holes* (small black areas) appear within the fields (large polygon almost completely filled in white) in Figure 4a and in Figure 4b. The pixels of those holes are not covered by the pattern covering the ones in the white areas. Their temporal behavior is thus different from their surrounding pixels though they should be related to the same crops. Some of those holes match pedological differences that have been reported by the experts while other holes are likely to be due to different fertilization and/or irrigation conditions. Such

information is particularly interesting as it can be used to adapt locally soil fertilization or irrigation.

Furthermore, it is possible to extract patterns corresponding to a single variety of a given crop. For example, with $\kappa = 5.5$, we have 1074 GFS-patterns, and 66 of them are maximal. Among these maximal ones, we have pattern $3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 1 \rightarrow 1 \rightarrow 1 \rightarrow 1$. Figure 5a gives its localization. While the previous pattern relates to wheat crop in general, that one relates to a particular variety. Indeed, 98.8% of the pixels it covers in the ground truth area are all of a same variety of wheat. Two rectangular fields are clearly identified (right part of the picture), the upper one corresponds to an area partially covered by the previous pattern, while this is not the case for the other rectangle, that exhibits another field of wheat. Both rectangles are covered by the general pattern corresponding to cultivated fields and shown Figure 4a.

The pixels covered by the patterns do not always correspond to cultivated areas, for instance, for $\kappa = 6$ we also obtained as a maximal GFS-pattern $2 \rightarrow 2 \rightarrow 2 \rightarrow 2 \rightarrow 2 \rightarrow 2 \rightarrow 2 \rightarrow 2$ that corresponds to paths, fallows, cities and field borders. Its localization is depicted in Figure 5b.

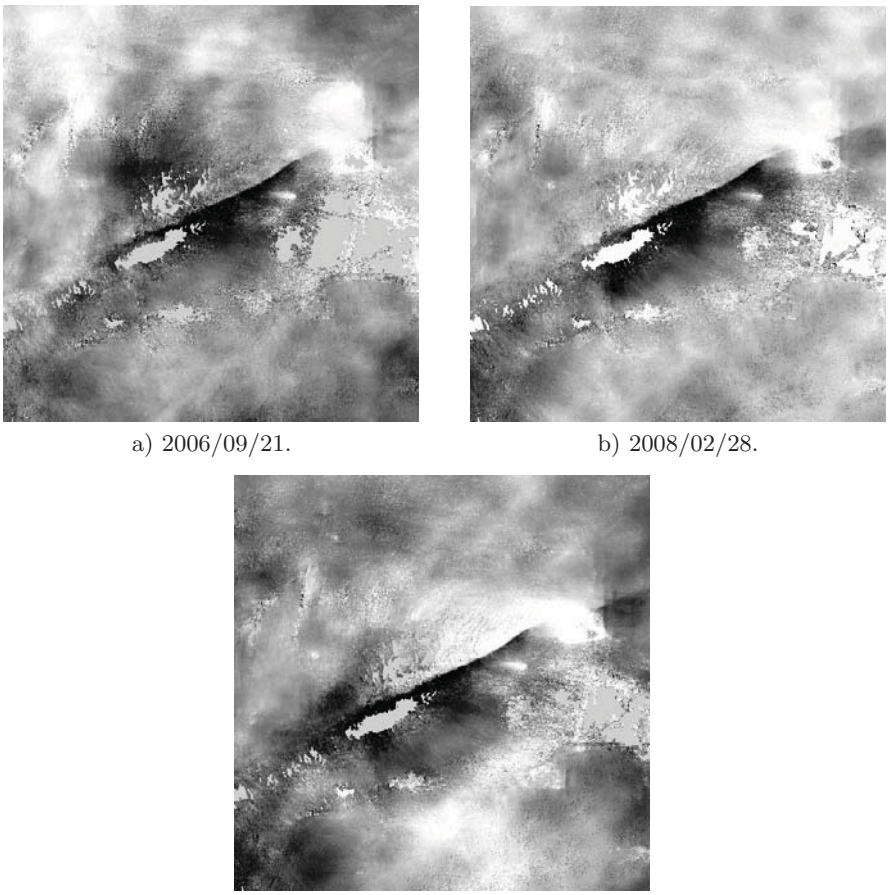
6 Handling a Radar SITS

In order to investigate the generality of the approach, we present other experiments on a very different type of SITS, a SAR (Synthetic Aperture Radar) SITS, while the ADAM SITS was an optical one. In addition, the spatial resolution is here of 80m $\times$ 80m per pixel and differs from the other SITS. Such a radar SITS is used in this section to find spatio-temporal ground displacement patterns. This kind of monitoring of ground deformation can be useful to plan long term soil usage (development of infrastructures for agriculture/transport).

This dataset and its preprocessing are presented in Section 6.1. The resulting dataset contains 491401 sequences of size 24. In Section 6.2, we show that the approach is useful to find meaningful patterns in such SAR interferometric data.

6.1 The SAR SITS: Presentation, Selection, and Preprocessing

Spaceborne SAR images are acquired by satellites that emit and record radar waves which are reflected by the earth surface. These radar images are referred to as *Synthetic Aperture Radar images* because of the signal processing technology. This technology allows to synthesize a 1 km wide radar antenna while the real radar antenna is only a 10 m wide one. In radar images, each pixel value is a complex number. Its magnitude (or module) relates to the amount of energy backscattered by the Earth's surface. Its phase (or angle) measures the propagation time of radar waves, i.e. the distance between the satellite and the Earth's surface, but it also includes a scattering term that relates to the nature of the target/ground surface. In order to use the geometrical information brought by



a) 2006/09/21. b) 2008/02/28.

c) 2009/08/06.

Fig. 6. Displacement (w.r.t the satellites) images at 3 different dates. Medium gray pixels represent short displacements, while the darkest and the clearest pixels respectively correspond to large negative and large positive values/displacements.

the phase, it is necessary to compute the phase difference between two SAR images acquired at different dates, under the hypothesis that the scattering term is stable: change due to temporal evolution and slightly different viewing angles should be limited. In this case, the resulting phase difference images, called *interferograms*, measure the distance difference which are related to the topography and possible Earth's surface displacement between the two acquisitions. By using *Digital Elevation Models (DEM)*, it is possible to remove the topographic component and obtain displacement measurements with a precision of a fraction of the wavelength (5.6 cm with the data used here). The main limitation of this so-called *Differential SAR Interferometry (D-InSAR)* comes from the variations

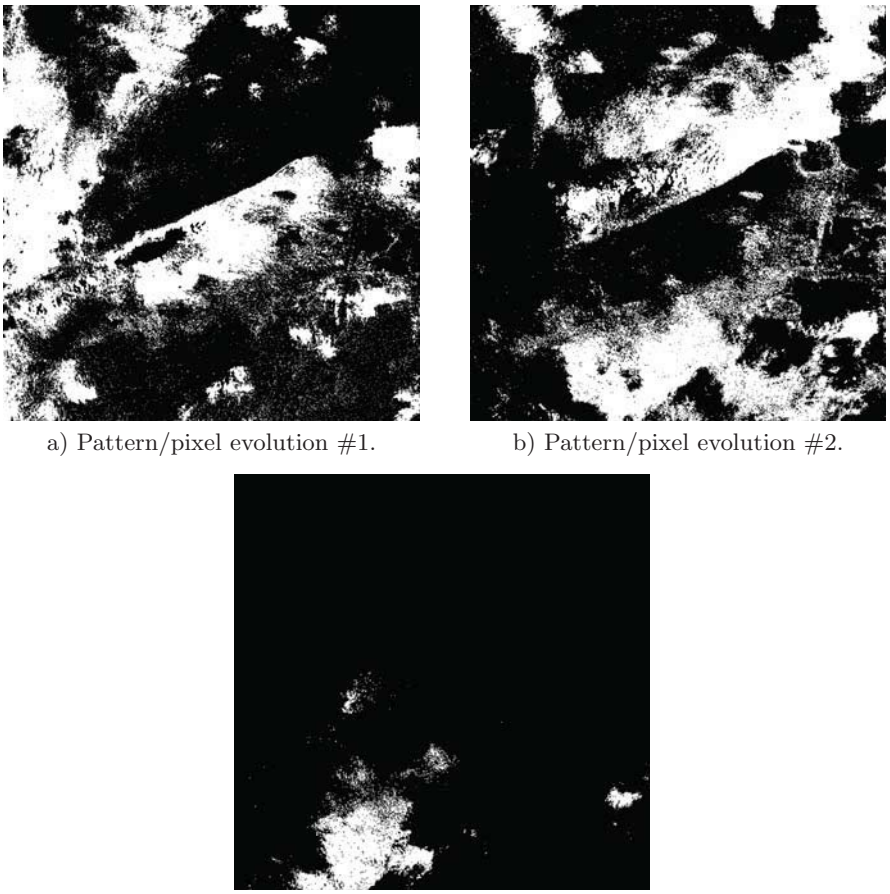
of the atmospheric conditions which may modify the propagation and introduce artifacts which are difficult to discriminate from the displacement information. The contribution due to the stratified atmosphere can be roughly estimated by using DEMs and meteorological data, but the effects of the turbulent atmosphere still degrade interferograms. Different approaches have been developed to reduce these difficulties by using interferogram time series: the *permanent scatterer (PS) technique* [34] which analyses the temporal signal on specific targets, the *Small BAseline Subsets (SBAS) strategy* [35] which selects the pairs of images that have been acquired with the most similar orbits (both in time and in space), or the STAMPS method [36] which incorporates both approaches. They are used by geophysicists to monitor ground surface deformations and their temporal evolutions.

In this experiment, a SAR SITS provided by the *Environmental Satellites (ENVISAT)* of the *European Space Agency (ESA)* is selected. It contains 25 images (701×701) acquired over the 2004-2009 period and covering the Haiyuan seismic fault in the north-eastern boundary of the Tibetan plateau. This area was affected by several major earthquakes in the early 20th century. The experts wish to locate and measure possible continuous crustal deformations though the acquisitions are affected by atmospheric perturbations. The selected SAR SITS is processed to derive displacement measurements by taking data characteristics into account without introducing any specific user knowledge of the studied ground deformation. First, interferograms are generated and then, residual orbital and atmospheric delays are removed using DEMs and meteorological data. In a third step, using a SBAS-based technique, the evolution of the phase between acquisition dates is computed. As a result, and since 25 SAR images were selected, an InSAR time series of 24 images containing the phase evolution/displacement between each acquisition dates is obtained. In other words, in these images, each pixel gives the displacement that has been observed between 2 different consecutive dates. As no smoothing is applied, both ground deformation and atmospheric turbulence contribute to the phase evolution/displacement. Typical images are shown for 3 different dates in Figure 6. The gray levels correspond to the displacements. Medium gray pixels represent short displacements, while the darkest and the clearest pixels respectively correspond to large negative and large positive values/displacements. Negative values/displacements indicate that the Earth's surface is getting closer to the satellite while the positive ones show that the Earth's surface is getting away from the satellite.

Finally, the whole dataset is quantified with 3 symbols ('1', '2' and '3') by using the 33rd and the 66th centiles. Symbol '1' represents large negative values, symbol '2' corresponds to low negative values and symbol '3' to positive values. The result is a base of sequences containing 491401 symbolic sequences of size 24.

6.2 Results

The quantified InSAR time series is mined to uncover relevant spatio-temporal structures. The goal is to extract GFS-patterns and present them to end-users



a) Pattern/pixel evolution #1. b) Pattern/pixel evolution #2.

c) Pattern/pixel evolution #3.

Fig. 7. Localization (white pixels) of 3 GFS-patterns. Pixels are a) getting closer to the satellite b) getting away from the satellite and c) getting away, then closer, then away again and finally closer to the satellite.

to draw attention to possible unknown displacements or displacement evolutions. The GFS-pattern extraction is performed by setting the minimum surface threshold σ to 4.07% (i.e. 20000 pixels) and the average connectivity threshold κ to 6. This leads to 3414 GFS-patterns. In order to focus on the most specific patterns, the maximal patterns having at least 10 symbols are selected. This leads to 19 patterns. For each one of these patterns, an image is built to observe the pixels that are affected by the evolution given by the pattern. This is illustrated by Figure 7, for the 3 following GFS-patterns:

- pattern #1: $2 \to 1 \to 1 \to 1 \to 1 \to 1 \to 1 \to 1 \to 1 \to 1$;
- pattern #2: $2 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 3 \to 2 \to 3$;

– and pattern #3: $3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 3 \rightarrow 1 \rightarrow 1 \rightarrow 3 \rightarrow 2 \rightarrow 3 \rightarrow 3 \rightarrow 2 \rightarrow 1$.

Pattern #1 indicates that some areas tend to get closer to the satellite while pattern #2 shows that other areas are getting away from the satellite. As it can be observed in Figure 7, these patterns are spatially complementary. A creep phenomenon is thus revealed by these two first patterns. It is coherent with the motion of the northern part of the studied zone (upper part of the image) that has been reported by the experts. Furthermore, the localization of the seismic fault that is due to this creep phenomenon can be inferred by looking at the crisp frontier between affected and non-affected pixels, especially on the lower-left to upper-right diagonal of the images. The third pattern, pattern #3, is indicating that the earth surface is getting away, then closer, then away again and finally closer to the satellite. This is quite an unusual behavior. In addition, the area that is affected is a very compact and isolated zone (see Figure 7 (c)). Therefore, civil engineering structures built in this zone should be designed so as to resist antagonist motions. The spatial localization of these three patterns shows that they enable to discriminate areas which are not easy to distinguish when looking at the images of the original SAR SITS (see Figure 6).

These results illustrate the potential of a combined use of advanced InSAR multi-temporal processing for deriving displacement time series and the GFS-pattern extraction for extracting, in an unsupervised way, spatio-temporal features corresponding to slow ground deformations. The refined SBAS approach is intended to reduce, as far as possible, systemic uncertainty, whereas the GFS-pattern offers end-users the possibility of exploring huge time series and discovering temporal evolutions which could be hidden by random uncertainty such as atmospheric turbulences. Finally, these results also show that the proposed method is quite generic as it can be applied to different types of datasets (optical data, radar data, different resolutions) for agricultural monitoring as well as for ground deformation monitoring for long term planning of soil usage.

7 Conclusion

In this paper, we applied the GFS-patterns to extract sets of pixels sharing similar evolution from Satellite Image Time Series over cultivated areas. Beside having a common temporal evolution, each one of these sets of pixels must be populated enough (support constraint) and connected enough (average connectivity constraint). We showed that the connectivity constraint can be partially pushed to prune the search space (along with the support constraint) and reduce significantly GFS-patterns extraction times. Furthermore, the parameter setting is intuitive and experiments on real data showed that using a simple maximality constraint allows to focus on small collections of patterns that are easy to browse and interpret. This makes the technique a good candidate for exploratory mining stages of agricultural Satellite Image Time Series.

The experiments also showed that, even on poor quality inputs (i.e., noisy images, rough quantization), the method can exhibit various level of details

of primary interest in agro-modelling (e.g., cultivated vs. non-cultivated areas, types of crops, varieties, ground deformations) and can be applied to both optical and radar Satellite Image Time Series. Even though our experiments were run in reasonable times on typical data, several directions to further improve the efficiency of the technique remain relevant. In particular, this includes investigating the use of an interval-based representation of occurrence locations (e.g., [37]) to speed-up occurrence handling.

Acknowledgments

The authors thank the French Research Agency (ANR) for supporting this work through the EFIDIR project (ANR-2007-MCDC0-04, www.efidir.fr) and the FOSTER project (ANR-2010-COSI-012-02, foster.univ-nc.nc). They also thank the ADAM project and the CNES agency for making the ADAM data available. The authors also wish to thank the European Space Agency (ESA) for the ENVISAT SAR data (Dragon project Dragon ID5305) over the Haiyuan fault. Finally, the authors express their gratitude to Roxana Vintila (Research Institute for Soil Science and Agrochemistry - Bucharest, Romania) and to Gheorghe Petcu (National Agricultural Research and Development Institute Fundulea, Romania) for supplying the ground truth of the regions that we studied through the ADAM project acquisitions.

References

1. Julea, A., Méger, N., Rigotti, C., Trouvé, E., Bolon, P., Lazarescu, V.: Mining pixel evolutions in satellite image time series for agricultural monitoring. In: Advances in Data Mining. Applications and Theoretical Aspects - 11th Industrial Conference, ICDM 2011, New York, NY, USA, August 30 - September 3, 2011. Proceedings. Lecture Notes in Computer Science, vol. 6870, pp. 189–203. Springer (2011)
2. Agrawal, R., Srikant, R.: Mining sequential patterns. In: Proc. of the 11th International Conference on Data Engineering (ICDE'95). pp. 3–14. IEEE Computer Society Press, Taipei, Taiwan (1995)
3. Julea, A., Meger, N., Bolon, P., Rigotti, C., Doin, M., Lasserre, C., Trouve, E., Lazarescu, V.: Unsupervised spatiotemporal mining of satellite image time series using grouped frequent sequential patterns. IEEE Transactions on Geoscience and Remote Sensing 49(4), 1417–1430 (2011)
4. Lillesand, T., Kiefer, R.: Remote Sensing and Image Interpretation. John Wiley and Sons, New York, fourth edn. (2000)
5. Héas, P., Datcu, M.: Modeling trajectory of dynamic clusters in image time-series for spatio-temporal reasoning. IEEE Transactions on Geoscience and Remote Sensing 43(7), 1635– 1647 (2005)
6. Honda, R., Konishi, O.: Temporal rule discovery for time-series satellite images and integration with RDB. In: Proc. of the 5th European Conference on Principles of Data Mining and Knowledge Discovery (PKDD'01). pp. 204–215. Freiburg, Germany (2001)
7. Mannila, H., Toivonen, H., Verkamo, A.I.: Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery 1(3), 259–289 (1997)

8. Nezry, E., Genovese, G., Solaas, G., Rémondière, S.: ERS - Based early estimation of crop areas in Europe during winter 1994-95. In: Proc. of the 2nd International Workshop on ERS Application. London (1995)
9. Gallucio, L., Michel, O., Comon, P.: Unsupervised clustering on multi-components datasets: Applications on images and astrophysics data. In: Proc. of the 16th European Signal Processing Conference (EUSIPCO'08). pp. 25–29. Lausanne, Switzerland (2008)
10. Ketterlin, A., Gançarski, P.: Sequence similarity and multi-date image segmentation. In: Proc. of the 4th International Workshop on the Analysis of Multitemporal Remote Sensing Images (MULTITEMP'07). pp. 1–4. Leuven, Belgium (2007)
11. Inglada, J., Favard, J.C., Yesou, H., Clandillon, S., Bestault, C.: Lava flow mapping during the Nyiragongo January, 2002 eruption over the city of Goma (D.R. Congo) in the frame of the international charter space and major disasters. In: Proc. of the IEEE International Geoscience and Remote Sensing Symposium (IGARSS'03). vol. 3, pp. 1540–1542. Toulouse, France (2003)
12. Vina, A., Echavarria, R., F., Rundquist: Satellite change detection analysis of deforestation rates and patterns along the colombia-ecuador border. AMBIO: A Journal of the Human Environment 33, 118–125 (2004)
13. Coppin, P., Jonckheere, I., Nackaerts, K., Muys, B., Lambin, E.: Digital change detection methods in ecosystem monitoring: a review. International Journal of Remote Sensing 25(9), 1565–1596 (2004)
14. Lu, D., Mausel, P., Brondizio, E., Moran, E.: Change detection techniques. International Journal of Remote Sensing 25(12), 2365–2407 (2004)
15. Li, L., Leung, M.: Robust change detection by fusing intensity and texture differences. In: Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'01). pp. 777–784. Kauai Marriott, Hawaii (2001)
16. Bontemps, S., Bogaert, P., Titeux, N., Defourny, P.: An object-based change detection method accounting for temporal dependences in time series with medium to coarse spatial resolution. Remote Sensing of Environment 112, 3181–3191 (2008)
17. Cao, H., Mamoulis, N., Cheung, D.W.: Mining frequent spatio-temporal sequential patterns. In: Proc. of the Fifth IEEE International Conference on Data Mining (ICDM'05). pp. 82–89. Houston, Texas, USA (2005)
18. Cao, H., Mamoulis, N., Cheung, D.W.: Discovery of periodic patterns in spatiotemporal sequences. IEEE Transaction on Knowledge and Data Engineering 19(4), 453–467 (2007)
19. Huang, Y., Zhang, L., Zhang, P.: A framework for mining sequential patterns from spatio-temporal event data sets. IEEE Transactions on Knowledge and Data Engineering 20(4), 433–448 (2008)
20. Nanni, M., Pedreschi, D.: Time-focused clustering of trajectories of moving objects. Journal of Intelligent Information Systems 27(3), 267–289 (2006)
21. Gudmundsson, J., Kreveld, M., Speckmann, B.: Efficient detection of patterns in 2D trajectories of moving points. Geoinformatica 11(2), 195–215 (2007)
22. Garofalakis, M., Rastogi, R., K., S.: Spirit: Sequential pattern mining with regular expression constraints. In: Proc. of the 25th International Conference on Very Large Databases (VLDB'99). pp. 223–234. Edinburgh, United Kingdom (1999)
23. Masseglia, F., Cathala, F., Poncelet, P.: The PSP approach for mining sequential patterns. In: Proc. of the 2nd European Conference on Principles of Data Mining and Knowledge Discovery in Databases (PKDD'98). pp. 176–184. LNAI 1510, Springer Verlag, Nantes, France (1998)

24. Pei, J., Han, B., Mortazavi-Asl, B., Pinto, H.: Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: Proc. of the 17th International Conference on Data Engineering (ICDE'01). pp. 215–226. Heidelberg, Germany (2001)
25. Pei, J., Han, J., Wang, W.: Constraint-based sequential pattern mining: the pattern-growth methods. Journal of Intelligent Information Systems 28(2), 133–160 (2007)
26. Srikant, R., Agrawal, R.: Mining sequential patterns: Generalizations and performance improvements. In: Proc. of the 5th International Conference on Extending Database Technology (EDBT'96). pp. 3–17. Avignon, France (1996)
27. Zaki, M.: Sequence mining in categorical domains: incorporating constraints. In: Proc. of the 9th International Conference on Information and Knowledge Management (CIKM'00). pp. 422–429. Washington, DC, USA (2000)
28. Zaki, M.: Spade: an efficient algorithm for mining frequent sequences. Machine Learning 42(1/2), 31–60 (2001)
29. Fisher, R., Dawson-Howe, K., Fitzgibbon, A., Robertson, C., Trucco, E.: Dictionary of Computer Vision and Image Processing. John Wiley and Sons, New York (2005)
30. Ng, R.T., Lakshmanan, L.V.S., Han, J., Pang, A.: Exploratory mining and pruning optimizations of constrained associations rules. In: Proc. of the ACM SIGMOD international conference on Management of data (SIGMOD'98). pp. 13–24. Seattle, Washington, USA (1998)
31. Pei, J., Han, J., Lakshmanan, L.V.: Mining frequent itemsets with convertible constraints. In: Proc. of the 17th International Conference on Data Engineering (ICDE'01). pp. 433–442. Heidelberg, Germany (2001)
32. Bonchi, F., Lucchese, C.: Pushing tougher constraints in frequent pattern mining. In: Proc. of the Pacific-Asia Knowledge Discovery and Data Mining conference (PAKDD'05). pp. 114–124 (2005)
33. Centre National d'Etudes Spatiales: Database for the Data Assimilation for Agro-Modeling (ADAM) project. Online (2010), http://kalideos.cnes.fr/index.php?id=accueil-adam
34. Ferretti, A., Prati, C., Rocca, F.: Permanent scatterers in SAR interferometry. IEEE Transactions on Geoscience and Remote Sensing 39(1), 8–20 (2001)
35. Berardino, P., Fornaro, G., Lanari, R., Sansosti, E.: A new algorithm for surface deformation monitoring based on small baseline differential SAR interferograms. IEEE Transactions on Geoscience and Remote Sensing 40, 2375–2382 (2002)
36. Hooper, A.: A multi-temporal InSAR method incorporating both persistent scatterer and small baseline approaches. Geophysical Research Letters 35 (2008)
37. Leleu, M., Rigotti, C., Boulicaut, J.F., Euvrard, G.: Constrained-based mining of sequential patterns over datasets with consecutive repetitions. In: Proc. of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'03). pp. 303–314. Springer-Verlag LNCS 2838, Cavtat-Dubrovnik, Croatia (2003)