Discovering Significant Evolution Patterns from Satelllite Image Time Series

François Petitjean, Florent Masseglia, Pierre Gancarski, Germain Forestier

To cite this version:

François Petitjean, Florent Masseglia, Pierre Gancarski, Germain Forestier. Discovering Significant Evolution Patterns from Satelllite Image Time Series. International Journal of Neural Systems, 2011, 21 (6), pp.15. 10.1142/S0129065711003024 . lirmm-00639480

HAL Id: lirmm-00639480
https://hal-lirmm.ccsd.cnrs.fr/lirmm-00639480v1

Submitted on 9 Nov 2011

DISCOVERING SIGNIFICANT EVOLUTION PATTERNS FROM SATELLITE IMAGE TIME SERIES

François PETITJEAN
LSIIT – UMR 7005, Pôle API
Bd Sébastien Brant – BP 10413
67412 Illkirch Cedex – France
E-mail: fpetitjean@unistra.fr

Florent MASSEGLIA
INRIA Sophia Antipolis
2004 route des lucioles – BP 93
06902 Sophia Antipolis – France
E-mail: florent.masseglia@sophia.inria.fr

Pierre GANÇARSKI
LSIIT – UMR 7005, Pôle API
Bd Sébastien Brant – BP 10413
67412 Illkirch Cedex – France
E-mail: gancarski@unistra.fr

Germain FORESTIER
LSIIT – UMR 7005, Pôle API
Bd Sébastien Brant – BP 10413
67412 Illkirch Cedex – France
E-mail: forestier@unistra.fr

Abstract

Satellite Image Time Series (SITS) provide us with precious information on land cover evolution. By studying these series of images we can both understand the changes of specific areas and discover global phenomena that spread over larger areas. Changes that can occur throughout the sensing time can spread over very long periods and may have different start time and end time depending on the location, which complicates the mining and the analysis of series of images. This work focuses on frequent sequential pattern mining (FSPM) methods, since this family of methods fits the above-mentioned issues. This family of methods consists of finding the most frequent evolution behaviors, and is actually able to extract long-term changes as well as short term ones, whenever the change may start and end. However, applying FSPM methods to SITS implies confronting two main challenges, related to the characteristics of SITS and the domain's constraints. First, satellite images associate multiple measures with a single pixel (the radiometric levels of different wavelengths corresponding to infra-red, red, etc.), which makes the search space multi-dimensional and thus requires specific mining algorithms. Furthermore, the non evolving regions, which are the vast majority and overwhelm the evolving ones, challenge the discovery of these patterns. We propose a SITS mining framework that enables discovery of these patterns despite these constraints and characteristics. Our proposal is inspired from FSPM and provides a relevant visualization principle. Experiments carried out on 35 images sensed over 20 years show the proposed approach makes it possible to extract relevant evolution behaviors.

1 Introduction

Change detection is an important field of remote sensing with various applications in land cover study. It has gained increasing attention due to recent important technological progress. The next generation of satellites (*e.g.*, $Ven\mu s$, *Sentinel-2*) will actually be able to acquire image time series at high temporal frequency. Satellite Image Time Series (SITS) are an important source of information for land cover analysis and change detection. Let us consider, for instance, the scene illustrated in Figure 1. Each image contains four main regions ($R1$ to $R4$) and the series show their evolution on four different dates. In July 2007, regions $R1$ and $R2$ were mainly covered with trees, region $R3$ with a river and region $R4$ with roofs. A global evolution to this example is the urbanization process, where the trees disappear, replaced by bare soil and finally by urban items (roads and roofs).

Extracting knowledge from time series can be crucial in domains such as prediction[1] and clustering[2, 3]. In the case of SITS, the discovery of evolutions has important applications[4, 5, 6] but requires to consider two important challenges related to the domain constraints and with SITS characteristics.

First, changes in a scene might spread over a long time period (urbanization, for instance, lasts for several years and building sites do not have the same start time and end time) or they might cycle (such as crop rotation). Let us consider the SITS illustrated in Figure 1. A possible region evolution from this series of images is the following: "trees → bare soil → urban", which can be observed for 50 % of pixels. Obviously, detecting such evolutions, having more than two steps, cannot rely on a naive comparison of two consecutive images. While the number of possible combinations over such long periods is very large, solutions to this kind of problem exist in knowledge discovery in the field of "frequent sequential pattern mining" (FSPM) [7, 8, 9]. In this paper, we will explore such solutions for change detection in SITS.

Second, satellites don't automatically give labels (such as "tree" or "roof") to (regions of) pixels. Actually, their sensors measure the light reflected by geographic areas for different wavelengths allowing satellites to acquire images on multiple bands, such as "Infra-red", "Red", "Green", etc. Let us now consider the images illustrated in Figure 1, from the satellite's point of view. Figure 2 shows the same series of images on the red band. In the image from July 2007, the red level of regions $R1$ and $R2$ could be 120 (which is the response on this band of some trees from broad-leaved or coniferous forests). The red level of region $R3$ is 30 (response of the river) and the red level of region $R4$ is 200 (response of red tile roofs in the red band). Meanwhile, broad-leaved trees and conifers have different responses on the Infra-Red band. Consequently, working with a single band is not sufficient since the number of possible values is lower than the number of possible kinds of surfaces.

This need for multiple bands is well known in geographic studies. Actually, a measure on a band generally takes a value in a range of 255 levels, whereas the number of possible measures for a pixel with n bands is 255^n, allowing much more possible combinations. Figure 3 shows the series illustrated in Figure 1, on the Infra-Red band. We can see, for instance, that the Infra-Red level is the same for red tile roofs and concrete roads.

Understanding from SITS the evolutions that occurred on a scene (*i.e.*, geographic area) is a complex task. Data mining methods are usually used rather than image/video processing methods, since SITS are very different from videos: (1) the images are not regularly sampled, (2) sensed values are absolute physical states of the land cover and (3) the series are much shorter. Many differences are then following in the conception of dedicated methods. The irregular sampling prevents making any assumptions on the content of previous/next images (frames in video). Moreover, satellite image time series are giving a physical state of the land cover; the method has thus to map the changes (as well as in videos), but also to characterize it in order to provide a semantic meaning to the thematic expert. Furthermore, video-dedicated methods have to be extremely fast in order to be able to process the data flow. We take here the opposite direction: the aim is to map and characterize the change with as few assumptions as possible.

Our goal is to extract patterns of evolutions from these images and, to that end, *we need to take into account the whole sets of dates and bands.* To date, the problem of mining frequent sequences from SITS has been investigated for a single band[10, 11]. Unfortunately, as explained above, a single band does not make it possible to distinguish between different kinds of surfaces.

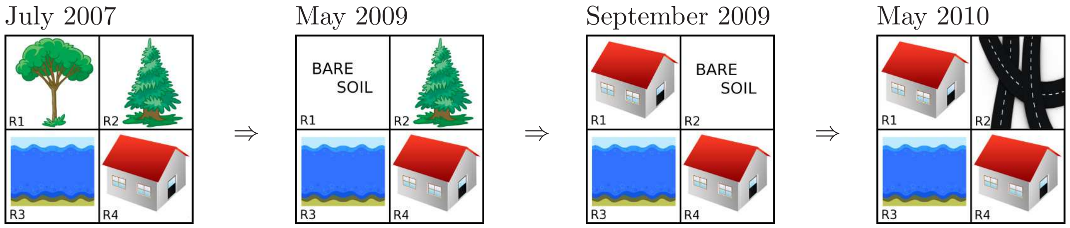


Figure 1: Example of change in a scene involving trees and urban growth where a pattern of evolution is "trees → bare soil → urban", verified by 50 % of regions (*i.e.* $R1$ and $R2$).

July 2007		May 2009		September 2009		May 2010	
level 120 *e.g.* conifer *e.g.* leafy	level 120 *e.g.* conifer *e.g.* leafy	level 170 *e.g.* soil	level 120 *e.g.* conifer *e.g.* leafy	level 200 *e.g.* roof	level 170 *e.g.* soil	level 200 *e.g.* roof	level 280 *e.g.* roads
level 30 *e.g.* river	level 200 *e.g.* roof	level 30 *e.g.* river	level 200 *e.g.* roof	level 30 *e.g.* river	level 200 *e.g.* roof	level 30 *e.g.* river	level 200 *e.g.* roof

Figure 2: The scene of Figure 1 from the "Red band" point of view

Let us consider a pattern discovery process on the red band, for instance. In this case we will extract region evolutions where broad-leaved trees and conifers have the same value but red tile roofs and concrete roads don't. Therefore, a possible frequent pattern would be "conifers/broad-leaved trees → bare soil". We can observe that "urban" does not appear since its surfaces take values in very different ranges on the red band. Actually, in the red band, the values of urban surface are divided into "roofs" ($R1$) and "roads" ($R2$). The expert would then interpret this pattern and decide that it can be reformulated as "trees → bare soil". Unfortunately, this pattern corresponds to the beginning of the urbanization process and we could not extract any information about "urban" surfaces in this analysis performed on the red band. The problem remains with other bands such as Infra-Red, for instance, where a possible pattern would be "bare soil/conifer → roof/road", reformulated by the expert as "bare soil → urban". In this case, we find the urban part of the pattern but it does not contain enough information about the trees (because they have different values on the Infra-Red band). Furthermore, the expert needs time for this interpretation (since the information on each step of the pattern is not sufficient) and the reformulation is not always reliable (because each value may correspond to multiple possible surfaces, making the result ambiguous).

Therefore, in order to find complete, significant and relevant patterns, we need to consider the whole set of bands. In our example, taking multiple bands into account would allow us to extract the pattern illustrated in Equation 1.

This pattern can easily be interpreted as "trees → soil → urban", which corresponds to our goal. It has two important advantages 1) it is more informative than the patterns extracted on a single band and 2) it is not ambiguous. As we will describe later, there is a strong correspondence between this problem and the problem of frequent sequential pattern mining (FSPM) [7, 8, 9, 12]. However, the use of frequent pattern mining methods for SITS analysis is not straightforward, since the non evolving regions, which are the vast majority, are overwhelming the evolving ones.

This article introduces a new strategy for the extraction of evolution patterns from SITS. The proposed method addresses the two problems of studying multi-spectral series and of extracting the main evolution behaviors from predominantly

$$\underbrace{(conifers/hardwoods)}_{Red} \rightarrow (\underbrace{soil}_{Red}, \underbrace{soil/conifers}_{Infra-Red}) \rightarrow \underbrace{(roof/road)}_{Infra-Red} \qquad (1)$$

July 2007		May 2009		September 2009		May 2010	
level 200 *e.g.* leafy	level 145 *e.g.* soil *e.g.* conifer	level 145 *e.g.* soil *e.g.* conifer	level 145 *e.g.* soil *e.g.* conifer	level 200 *e.g.* roof *e.g.* road	level 145 *e.g.* soil *e.g.* conifer	level 200 *e.g.* roof *e.g.* road	level 200 *e.g.* roof *e.g.* road
level 20 *e.g.* river	level 200 *e.g.* roof *e.g.* road	level 20 *e.g.* river	level 200 *e.g.* roof *e.g.* road	level 20 *e.g.* river	level 200 *e.g.* roof *e.g.* road	level 20 *e.g.* river	level 200 *e.g.* roof *e.g.* road

Figure 3: The scene of Figure 1 from the "Infra-Red band" point of view

static behaviors. In this paper, we propose to analyze a scene with satellite images on important time periods. Our approach will be tested over 35 images and a period of 20 years, which represents 28 million sensed values.

This paper is organized as follows. In Section 2 we give an overview of existing works in SITS analysis. Section 3 gives the main definitions of frequent sequential patterns and Section 4 describes the preprocessing of SITS for the discovery of such patterns. In Section 5 we propose a FSPM method devoted to SITS, along with a visualization principle, and our results are described in Section 6. Finally, we conclude this paper in Section 7.

2 Related Works: SITS Analysis

Change detection in a scene allows the analysis, through observations, of land phenomenon with a broad range of applications such as the study of land-cover or even the mapping of damages following a natural disaster. These changes may be of different types, origins and durations.

In the literature, we find three main families of change detection methods. **Bi-temporal analysis**, *i.e.*, the study of transitions, can locate and study abrupt changes occurring between two observations. Bi-temporal methods include image differencing[13], image ratioing[14] and change vector analysis (CVA)[15]. A second family of **mixed methods**, mainly statistical methods, applies to two or more images. They include methods such as post-classification comparison[16], linear data transformation (Principal Component Analysis and Maximum Autocorrelation Factor)[17], image regression or interpolation[18] and frequency analysis (*e.g.*, Fourier, wavelet)[19]. Finally, we find methods designed towards image time series and based on **radiometric trajectory analysis**[20]. Whatever the type of methods used in order to analyze satellite image time series, there is a gap between the amount of data representing these time series, and the ability of algorithms to analyze them. First, these algorithms are often dedicated to the study of a change in a scene from bi-temporal representation. Second, even if they can map change areas they are not able to characterize them. As for multi-date methods, their results are usually hard to interpret and do not characterize the change.

FSPM [7, 8] is for its part intending to extract patterns of evolution in a series of symbols. These methods enable to identify sets of sequences that had the same underlying evolution. Furthermore, they are able to characterize this evolution, by extracting the pattern shared by this set of sequences. Extracting frequent sequences from SITS was introduced in[10, 11]. The authors study the advantages of such sequences in two applications: weather and agronomics. However, their proposal is restricted to the mining of series of images where each pixel take a value on a single band only. Our proposal, as explained in Section 4, applies to images where the pixels take values on tuples, each value corresponding to a separate band. These characteristics, along with the large number of images, will have important consequences on the patterns, on their relevance and on the complexity of their discovery.

3 Mining frequent sequential patterns

Sequential patterns are extracted from large sets of records. These records contain sequences of values that belong to a specific set of symbols, as stated by Definition 1 (inspired by the definitions of[7]).

Definition 1 *Let* $\mathcal{I} = \{i_1, i_2, ..., i_m\}$*, be a set of* m *values (or* items*).* $\mathcal{I}$ *is a vocabulary of all possible values for an item. Let* $I = \{t_1, t_2, ...t_n\}$*, be a subset of* $\mathcal{I}$*.* I *is called an* itemset *and is noted* $(t_1; t_2; ...; t_n)$*. A* sequence s *is a non-empty list* *of itemsets noted $< s_1, s_2, \ldots, s_n >$ where s_j is an itemset.* A data sequence *is a sequence in the dataset being analyzed.*

Definition 2 shows the conditions for the inclusion of two sequences. In other words, s_1 is included in s_2 if each itemset of s_1 is included in an itemset of s_2 with the same order. This definition is illustrated by examples 1 and 2.

Definition 2 *Let $s_1 =< a_1, a_2, \ldots, a_n >$ and $s_2 =< b_1, b_2, \ldots, b_m >$ be two sequences. s_1 is* included *in s_2 ($s_1 \prec s_2$) if and only if $\exists i_1 < i_2 < \ldots < i_n$ integers, such that $a_1 \subseteq b_{i_1}$, $a_2 \subseteq b_{i_2}, \ldots,\ a_n \subseteq b_{i_n}$.*

Example 1 *Let us consider the sequence $s_1 =< (3)\ (4;5)\ (8) >$. s_1 is made of 3 ordered itemsets. The first itemset of s_1 contains only one item (i.e. (3)). The second itemset of s_1 contains two items (i.e. $(4;5)$ an itemset of size two that contains the items 4 and 5). The third itemset of s_1 contains only one item (i.e. (8)). Let us now considered the sequence $s_2 =< (7)\ (3;8)\ (9)\ (4;5;6)(8) >$. Then, s_1 is included in s_2 (i.e $s_1 \prec s_2$) since $(3) \subseteq (3;8)$, $(4;5) \subseteq (4;5;6)$ and $(8) \subseteq (8)$. Meanwhile, the sequence $s_3 =< (3;8;9)\ (4;5) >$ is not included in s_2 since $(3;8;9)$ is not included in an itemset of s_2.*

Example 2 gives an illustration of how sequential pattern mining may be applied to SITS as a model of their content and meaning.

Example 2 *Let us consider s_1 from Example 1. Say that 3 stands for "low IR", 4 corresponds to "low R", 5 to "Average NDVI" and 8 to "high NDVI". Then, if a pixel has a series of values that corresponds to s_1, it should be regarded has a pixel having a low value of infra-red in an image, followed by a low value of red and an average value of NDVI in a next image and lastly by a high value of NDVI in a next image.*

In this paper, the main characteristic for sequential pattern extraction will be the frequency of the patterns. This notion is based on the number of occurrences of a pattern, compared to the total number of sequences, as stated by Definition 3. Finally, for simplicity in the results, only the longest patterns are kept (C.f. Definition 4).

Definition 3 *A data sequence s_d* supports *a sequence s (or participates in the support of s) if $s \prec s_d$. Let D be a set of data sequences. The* support *of s in D is the fraction of data sequences in D that support s:*

$$support(s) = \frac{|\{s_d \in D | s \prec s_d\}|}{|D|} \qquad (2)$$

Let minSupp be the minimum support *value, given by the end-user. A sequence having support higher than minSupp is frequent.*

Example 3 *Let us consider the toy database given in Table 1 that contain four sequences. If the end-user gives a minimum support of 50%, then a sequence will be frequent if it is supported by (i.e. included in) at least two sequences of this database. This is the case of $< (a)\ (b)\ (c;x) >$, which is included in three sequences among three, thus having a support of 75% (above the minimum support given by the user).*

Definition 4 *Let F^D be the set of frequent sequential patterns in D. In a set of sequences, a sequence s is maximal if s is not contained in any other sequence. Let L^D be the set of maximal sequences of F^D. L^D is the set of* maximal frequent sequential patterns *in D.*

Table 1: Sample database

s_1	$< (a)\ (b;d)\ (c;e;x) >$
s_2	$< (a;b)\ (b)\ (c;f;x) >$
s_3	$< (n;o)\ (p)\ (q;s) >$
s_4	$< (a;g)\ (b;m)\ (c;g;x) >$

4 Data preparation (SITS)

We want to analyze images from SITS databases, such as Kalideos.[*] In our experiments, for instance, we have extracted a series of 35 images Spot-1, Spot-2 and Spot-4 (the scenes are located in the south-west of France) as illustrated by Figure 4. These images are described by three attributes, corresponding to three spectral bands: Near Infra-Red (NIR), Red (R) and Green (G). To these bands, we add a fourth one, corresponding to the Normalized Difference Vegetation Index (NDVI). It is the most used index in remote sensing since it makes it possible to distinguish between several spectral signatures. The NDVI is calculated as follows for a pixel p: $\text{NDVI}\,(p) = \frac{\text{NIR}(p) - \text{R}(p)}{\text{NIR}(p) + \text{R}(p)}$.

Since the images from these databases are acquired by different sensors, the comparison of radiometric levels of a pixel (x, y) from one image to another calls for corrections. The value of each pixel has to be adjusted. First, we need to make sure that pixel (x, y) in a series covers the very same geographic area in every image. Then, some corrections are performed in order to reduce the impact of the difference of the solar lighting

[*] http://kalideos.cnes.fr

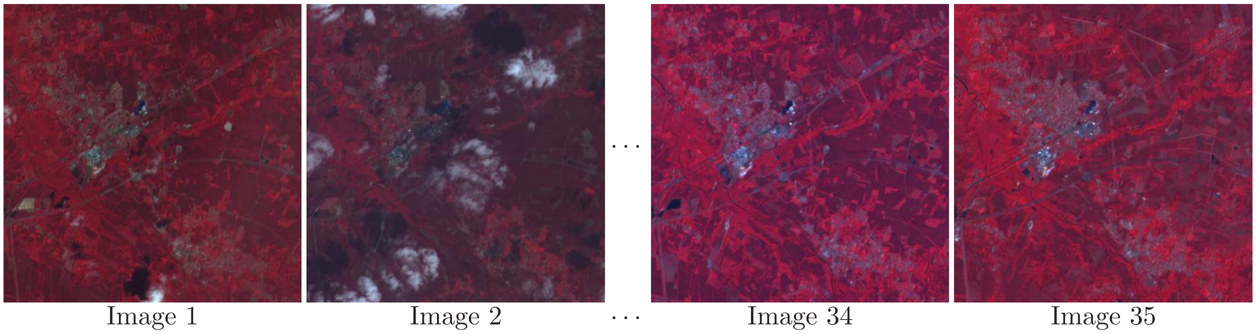


Figure 4: Extract of the Satellite Image Time Series from the Kalideos database. CNES 2010 – Distribution Spot Image

and of the difference of composition of the atmosphere. These corrections make it possible to guarantee that a geographic area (x, y) can be studied throughout the image series with comparable values.

This section formalizes the construction of the sequences from the image series. The idea is to study the evolution of atomic geographic areas identified by their (x, y) coordinates. In this way, the image series provide, for each sensed area (x, y), a 4-dimensional series of values (Definitions 5 and 6. Then, the values have to be quantified in order to be studied by the frequent pattern mining method (Definition 7).

Notations Let us first introduce several notations:

- $\llbracket X, Y \rrbracket = [X, Y] \subset \mathbb{N}$: the set of integer values from X to Y;
- $\{\!\{a, a, c\}\!\}$ refers to the multi-set theory, *i.e.*, sets with possibly multiple occurrences of elements;
- $\uplus$ refers to the union operator for multi-sets, *e.g.*, $\{\!\{a\}\!\} \uplus \{\!\{a\}\!\} = \{\!\{a, a\}\!\}$;
- $I_b^n(x, y)$ denotes the reflectance level (sensed value) of the n^{th} image in the b^{th} band.

Definition 5 *Let $\mathcal{S}_{image} =< I^1, \ldots, I^{\mathcal{N}} >$ be a series of $\mathcal{N}$ images of width $\mathcal{W}$ and eight $\mathcal{H}$. Let $\mathcal{B}$ be the number of bands in the images. Each multi-valued (with multiple bands) image I^n $(n \in [1, \mathcal{N}])$ can be seen as a function:*

$$\begin{array}{llll} I^n : & \llbracket 1, \mathcal{W} \rrbracket \times \llbracket 1, \mathcal{H} \rrbracket \rightarrow & & \mathbb{Z}^{\mathcal{B}} \\ & (x, y) & \mapsto I^k(x, y) = & \prod_{b=1}^{\mathcal{B}} I_b^n(x, y) \end{array} \tag{3}$$

with $\prod$ be the Cartesian product.

Definition 6 *Let $\mathcal{S}^v$ be the dataset built from the values of the image series. $\mathcal{S}^v$ is the set of sequences defined as:*

$$\begin{array}{r} \mathcal{S}^v = \{\!\{< I^1(x, y), \cdots, I^{\mathcal{N}}(x, y) > \\ | x \in \llbracket 1, \mathcal{W} \rrbracket, y \in \llbracket 1, \mathcal{H} \rrbracket \}\!\} \end{array} \tag{4}$$

Finally, a discretization step is necessary on the values of bands for a sequential pattern extraction. Actually, this step will lower the total number of items during the mining step. To that end, we first create $\mathcal{B}$ datasets D_b $_{(1 \leqslant b \leqslant \mathcal{B})}$: one dataset per band from all images. Then, in each such dataset, we gather together all the values of the corresponding attribute. It makes it possible to have a common discretization of the values between all images. In this way, the first slice of NIR is for example the same whatever the considered quantified image. Formally, $\mathcal{B}$ datasets of non-temporal data are created as:

$$D_b = \forall (x, y) \in \llbracket 1, \mathcal{W} \rrbracket \times \llbracket 1, \mathcal{H} \rrbracket : \biguplus_{n=1}^{\mathcal{N}} I_b^n(x, y) \tag{5}$$

Then, to each dataset D_i, we apply the K-means algorithm [21] in order to obtain K groups of values. Although all discretization techniques can be used here, the K-means algorithm was chosen instead of a usual histogram equalization, since the K-means algorithm is more adaptive to the distribution of the data. Therefore, every attribute (NIR, R, G or NDVI) is divided into K groups. For readability, we name these groups ordered by the value of the centroid (*i.e.*, the mean value of the group). For instance, the cluster NIR_1 corresponds to the first slice of NIR, *i.e.*, the cluster of NIR with lowest values, NIR_K corresponds to the last slice of NIR, *i.e.*, the cluster of NIR with highest values. Definition 7 details the creation of $\mathcal{S}$: the dataset ready for sequential pattern extraction, from the values of $\mathcal{S}^v$.

Definition 7 *Let Clus be the function associating the $\mathcal{B}$ values of a pixel (*i.e., *a quadruplet $(nir, r, g, ndvi)$) with their $\mathcal{B}$ slices computed by the* K-MEANS *algorithm*[†]*. The dataset $\mathcal{S}$ from the discretized values of the image series is then defined as:*

$$\mathcal{S} = \{\!\{ < Clus(I^1(x,y)), \cdots, Clus(I^{\mathcal{N}}(x,y)) > \\ | x \in [\![1, \mathcal{W}]\!], y \in [\![1, \mathcal{H}]\!] \}\!\} \quad (6)$$

We are thus provided, for each pixel, with a sequence of discrete values as, for example: $(NIR_1; R_6; G_3; NDVI_{16}) \rightarrow \cdots \rightarrow (NIR_{12}; R_3; G_{14}; NDVI_{19})$ where $(NIR_1; R_6; G_3; NDVI_{16})$ means that the value of that pixel in the first image is in the first slice of near infra-red, in the sixth slice of red, in the third slice of green and in the 16th slice of $NDVI$.

5 Extracting and Visualizing Patterns from SITS

The preprocessing steps described in Section 4 provide us with a series of images where each pixel is described on a tuple of values. Let us consider the series of three images simply reduced to 4 pixels ($p1$ to $p4$) illustrated by Figure 5. We want to extract frequent evolutions from these images. In other words if there exists a large enough set of pixels with the same "behavior" (*i.e.* these pixels have the same evolution), then this behavior must be discovered. Let us mention that the pixels' position is not a criterion here (our goal is not to extract pixels because of a shape). Our goal is to extract significant schemes in the evolution of a set of pixels. Each pixel in this figure is described on 3 values (corresponding to bands $B1$ to $B3$). With a minimum support of 100 %, there is no frequent pattern in these images (no "behavior" corresponding to the whole set of pixels). With a minimum support of 50 %, however, we find two frequent behaviors:

1. $< (B1, white; B2, white)\ (B1, gray; B2, red) >$. This behavior matches the sequences of values of pixels $p2$ on images 1 and 2 (or 3) and $p3$ on images 1 (or 2) and 3.
2. $< (B1, white; B2, white)\ (B1, white; B2, white) >$. This behavior matches the sequences of values of pixels $p1$ (let $p1$ be white on all images) $B2$ on on images 1 and 2 and $p3$ on images 1 and 2.

Let us note that, in the illustration above, patterns may be frequent even despite a lag in the images that support them (according to Definition 3).

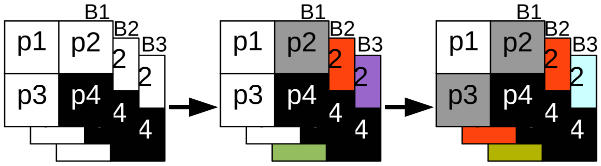


Figure 5: A series of three images, with four pixels described on three bands.

5.1 Mining Sequential Patterns from Image Time Series

As described above, our goal is to extract sequential patterns from these image time series. However, the pixels that support the evolution patterns embedded in SITS are not the vast majority. In this context, the sequential patterns having the highest support will usually correspond to "non-evolution patterns." In other words, the sequences in these very frequent patterns contain the same frequent item, repeated several times. An example of such a pattern being $< (trees)(trees)(trees) >$ meaning that, in this series, most pixels have a value corresponding to trees on images i, j and k (with $i < j < k$). This would correspond to $< (B1, white; B2, white)\ (B1, white; B2, white) >$ in our previous illustration, which means that the *white* value was found in the two attributes $B1$ and $B2$ at the same time, and that another same state was found afterwards.

However, such patterns are not really informative. In SITS, patterns with the highest support always correspond to geographic areas that did not change and these areas are the vast majority. However, as illustrated in the introduction, mining frequent sequential patterns from satellite images can reveal pixels having the same evolution (since these pixels will be characterized by the same pattern). *Therefore, the challenge is to extract patterns having a minimum threshold, while avoiding the enumeration of non-evolving patterns.* There are several naive approaches for solving this problem, but they all have important drawbacks.

A first idea would be to remove the pixels that never show any change. This is not enough, since the non-evolution part of the remaining pixels might still cause the same problems. Actually, a pixel might not show any change for years, and some important changes at the end of the series.

[†] For instance, a pixel taking values $(12; 35; 200; 100)$ in an image could have $Clus(12; 35; 200; 500) = (NIR_1; R_2; G_{16}; NDVI_{11})$.

Another approach would consist in lowering the minimum support until we obtain patterns that correspond to evolutions (since the areas of changes are minority). This would lead to extracting patterns having lower support (say, above 2 %). Unfortunately, in this case, the whole (and large) set of frequent values will flood the process with an intractable number of candidate and frequent patterns with two important consequences. First, the results are difficult, or even impossible, to obtain (because of the combinatorial complexity associated with frequent pattern extraction). Second, even if the results could be obtained, they would be difficult to read and very few would be relevant because they would contain a lot of non-evolution patterns.

A third idea would consist of a candidate generation process where two successive occurrences of the same value is forbidden. Indeed, such a solution would be motivated by the need to get rid of non-evolution patterns (such as $<(trees)(trees)(trees)>$). Unfortunately, this solution is not sufficient because non-evolution patterns can take values on the whole set of available bands. Let us consider, for instance, the pattern $<(tree, b1)(tree, b2)(tree, b3)>$. This pattern does not contain any repetition of the same value since $(tree, b1)$ corresponds to the response of trees on band 1, $(tree, b2)$ on band 2 etc. Therefore, since $(tree, b1) \neq (tree, b2) \neq (tree, b3)$, this pattern would be considered.

Finally, a naive and extreme solution would consist in the total removal of highly frequent items from the candidate generation process. In this case, we could ignore important patterns that should be considered. Let us consider the case where $(tree, b1)$ and $(tree, b2)$ are very frequent, whereas the item $(urban, b3)$ is barely frequent. Then, the pattern $<(tree, b1)(tree, b2)>$ would be ignored (which is correct) but also the pattern $<(tree, b1; tree, b2)(urban, b3)>$ (which must not be the case).

Considering the number of naive (and unrealistic) solutions, we propose a frequent pattern extraction algorithm based on PSP[8][‡]with a targeted adjustment: the candidate generation step discards patterns having two contiguous highly frequent items. To that end, we introduce a maximum threshold (described in Definition 8) in addition to the minimum support of Definition 3. Besides the ability to discard non-evolution patterns, this principle significantly reduces the combinatorial complexity associated with FSPM, as we explain in the following analysis.

Definition 8 *Let $maxSupp$ be a maximum frequency threshold and H be the set of items having support larger than $maxSupp$ ($H = \{i \in \mathcal{I} | support(i) > maxSupp\}$). Let $minSupp$ be a minimum frequency threshold and FI be the set of items having support larger that $minSupp$. The* reduction factor *h that applies to FI according to $maxSupp$ is given by $h = |FI|/|FI \backslash H|$.*

It is straightforward to show that a lower number of combinations leads to a lower number of possible frequent sequential patterns. However, with Theorem 1 we provide a formal comparative analysis of the upper bounds on the number of possible frequent sequences when successive occurrences of highly frequent items are considered and when they are discarded. For simplicity (and to make the upper bounds easier to read while remaining reliable) we consider that the empty itemset is not discarded from the set of possible k-itemsets, $\forall k$. Therefore, the sequences containing one or more occurrences of the empty itemset are not discarded from the following reasoning. For instance, sequences like $<(1)\ ()\ (4)\ ()\ (2)>$, which contains two occurrences of the empty itemset, are not considered is this reasoning. Actually, the impact of this itemset (and its repetitions) on the possible number of sequences is negligible compared to the exponential number of potential combinations involved by sequential patterns. Furthermore, our reasoning is limited to sequences of pair length (though it can be generalized to any length of sequences).

Theorem 1 *Let n be the number of frequent items and h be the reduction factor introduced in Definition 8. Let S_t be the set of possible sequences of length t and $P_{[0..t]} = \bigcup_{j=0}^{t} S_t$ be the set of possible sequences of length 0 to t. Let S'_t be the set of possible sequences of length t with no contiguous successive occurrences of frequent items and $P'_{[0..t]} = \bigcup_{j=0}^{t} S'_t$ be the set of possible such sequences of length 0 to t. Then $|P_{[0..t]}| = \sum_{j=0}^{t} 2^{nt}$ and $|P'_{[0..t]}| = \sum_{j=0}^{t} \frac{2^{nt}}{h^{\frac{t}{2}}}$ (and $|P'_{[0..t]}| < |P_{[0..t]}|$).*

Proof. First, let us recall some properties of sequential patterns. $|PI|$, the number of possible itemsets with n frequent items is given by the number of itemsets having length 0 plus the number of itemsets having length 1, etc. The number of itemsets having length k is given by C_n^k (*i.e.* the number of choices of k items among n possible items in $\mathcal{I}$). Therefore, $|PI| = \sum_{k=0}^{n} C_n^k = 2^n$ (or $(2^n - 1)$ if we discard the empty itemset). $|S_t|$, the number of possible sequences of length t is given by $|S_t| = 2^{nt}$. Actually, the number of possible combinations for S_t is given by:

$$\underbrace{2^n \times 2^n \times \ldots \times 2^n}_{t\ times} \tag{7}$$

[‡] However, this principle may apply to any sequential pattern mining algorithm.

And $|P_{[0..t]}|$, the number of possible sequential patterns of length 0 to t is given by:

$$|P_{[0..t]}| = |\bigcup_{j=0}^{t} S_t| = \sum_{0}^{t} 2^{nt} \tag{8}$$

Second, let us give an upper bound on the number of sequential patterns when two items of H cannot be combined. $|PI \backslash H|$, the number of possible itemsets i such that $minSupp < support(i) < maxSupp$ is given by $|PI \backslash H| = \frac{|PI|}{h} = \frac{2^n}{h}$. The number of possible combinations for S_t, the sequences of length t with no contiguous successive occurrences of items from H is given by:

$$\underbrace{2^n \times \frac{2^n}{h} \times 2^n \times \frac{2^n}{h} \times \ldots \times 2^n \times \frac{2^n}{h}}_{t\ times} \tag{9}$$

Therefore, $|S'_t| = (2^n \times \frac{2^n}{h})^{\frac{t}{2}} = \frac{2^{2n\frac{t}{2}}}{h^{\frac{t}{2}}} = \frac{2^{nt}}{h^{\frac{t}{2}}}$ and the number of possible such sequential patterns of length 0 to t is given by:

$$|P'_{[0..t]}| = |\bigcup_{j=0}^{t} S'_t| = \sum_{0}^{t} \frac{2^{nt}}{h^{\frac{t}{2}}} \tag{10}$$

□

The importance of h is significant regarding the number of possible itemsets. However, when it comes to sequential patterns, it becomes crucial. Figure 6 gives the difference between the upper bounds on itemsets and sequential patterns with 8 to 10 frequent items, a reduction factor (h) between 1.5 and 5, and a sequential pattern length (t). With $h = 1.5$, the maximum number of itemsets (left diagram in Figure 6) drops from approximately 1000 to 600, which is expected to have a limited influence on the frequent itemset discovery process. Meanwhile, the maximum number of sequential patterns of size t (right diagram) shows a very large theoretical difference, growing up to an order of magnitude for $h = 1.5$ (and even more for higher values of h).

5.2 Towards a generic learning approach for SITS mining

The new principle of FSPM presented in the last subsection is inspired from the constraints of SITS. The resulting algorithm PSP_{SITS} is detailed in Algorithm 1. It is based on PSP, while providing a new threshold value standing for the maximum frequency.

Algorithm 1 PSP_{SITS}

```
Input: $D$   //a database of sequences
Input: $I$   //the set of items in D
Input: $minSupp$   //the minimum support
Input: $maxSupp$   //the maximum support
Output: $k$   //the maximum length of a frequent
Output: $L_{0,k}$   //the set of frequent sequences of length 0 to k
  $k \leftarrow 1$
  //Extract the frequent items of D
  $L_k \leftarrow \{\ i \mid i \in I\ ,\ support(i) > minSupp\}$
  //Index the highly frequent items of D
  $H \leftarrow \{\ i \mid i \in I\ ,\ support(i) > maxSupp\}$
  $k \leftarrow k + 1$
  $C_k \leftarrow$ Learning$(L_{k-1}, H)$
  repeat
    //Check the support of each candidate D
    CountSupport$(C_k, D)$
    //Extract the frequent ones
    $L_k =$ FreqSeq$(C_k)$
    $k \leftarrow 1$
    //Generate the next step candidates
    if $L_k \neq \emptyset$ then
      $C_k \leftarrow$ Learning$(L_{k-1}, H)$
    end if
  until $L_k = \emptyset$
```

The main modification of PSP that should be noticed in PSP_{SITS} is the usage of a maximum support in the learning step, where the candidates of each level are generated. We do not give the details of this candidate generation algorithm because, above this particular algorithm, an important characteristic of our work is that it can apply to any FSPM method. Our theoretical background shows that the number of generated candidates, during the learning phase, gives a very low number of candidate patterns that have to be checked over the data (compared to traditional learning principles). This is due to the fact that candidates containing two consecutive items having support greater than $maxSupp$ (see algorithm PSP_{SITS} in Algorithm 1) will be rejected. This principle is very easy to apply to any FSPM algorithm by modifying the learning step. For instance, in GSP [7], the learning step is based on a pattern matching principle that checks the subsequences of each couple of frequent sequences. When the subsequence obtained by removing the first item of the first sequence matches the subsequence obtained by removing the last item of the second sequence, then a candidate is generated. For instance, let $s_1 =< (a)(bc)(d) >$ and

$s_2 =< (bc)(d)(e) >$ be two frequent sequences having length 4, then we can generate the candidate sequence $c =< (a)(bc)(d)(e) >$ (because the subsequences of s_1 and s_2 do match). If we want to apply our filter on the consecutive highly frequent items, all we have to do is to carefully generate the candidates of size 2 and 3. Afterwards, the property of not having two consecutive highly frequent itemsets in a sequence will be automatically propagated without any additional effort.

5.3 Visualizing Sequential Patterns Extracted from SITS

Our goal, in this section, is to propose a visualization approach designed towards the synthetic representation of sequential patterns in SITS.

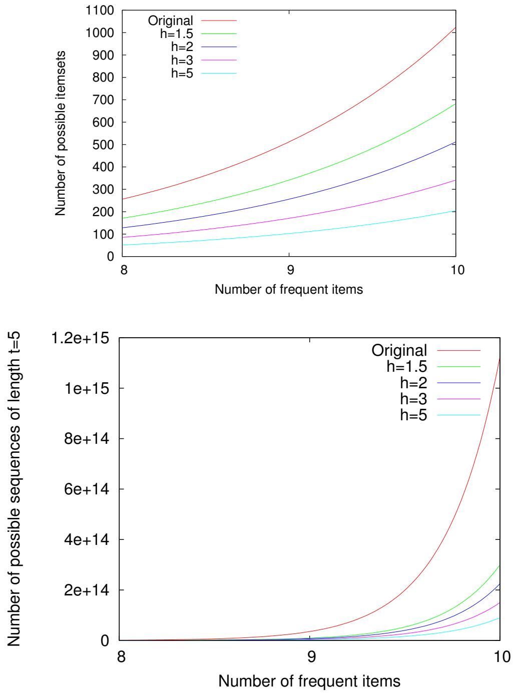


Figure 6: The theoretical influence of h is much more significant on the number of possible sequential patterns than it is on the number of possible itemsets.

The (usually very) large number of patterns extracted by means of data mining algorithms is an important issue, with numerous and relevant contributions in the literature[22, 23, 24]. In the case of patterns extracted from SITS, an important advantage is that our individuals are pixels. The main idea is that every sequence (as introduced in Definition 7) built from the SITS corresponds to a geographic location identified by its coordinates in a $\mathcal{H} \times \mathcal{W}$ space (with $\mathcal{H}$ and $\mathcal{W}$ be the height and width of the images in the series). Therefore, we can build a $\mathcal{H} \times \mathcal{W}$ image, where the data sequences that participate to the support of one or more frequent sequential patterns will be highlighted (since each data sequence corresponds to a pixel in that space). Moreover, by assigning gradual levels to the pixels, we can see the percentage of frequent sequences they support. To the best of our knowledge, *this is the first visualization principle intending to give a synthetic representation of the set of frequent sequential patterns extracted from a series of images.* In Definition 9, we define the "contribution frequency" of a data sequence in a set of sequential patterns. The larger the number of patterns supported by a data sequence, the higher its contribution frequency.

Definition 9 *Let $F^{\mathcal{S}}$ be the set of frequent sequential patterns in $\mathcal{S}$. The contribution frequency of a sequence $s_{x,y} =< Clus(I^1(x,y)), \cdots, Clus(I^n(x,y)) >$ in $F^{\mathcal{S}}$ is defined as:*

$$\mathcal{C}(s_{x,y}) = \frac{|\{s_F \prec s_{x,y} \mid s_F \in F^{\mathcal{S}}\}|}{|F^{\mathcal{S}}|} \qquad (11)$$

Definition 10 *Let $\mathcal{C}(s_{x,y})$ be the contribution frequency of a sequence $s_{x,y}$. The contribution frequency image is the image gathering the contribution frequency of all sequences identified by their coordinates (x,y). $I_{\mathcal{C}}$ is defined as:*

$$\begin{array}{llll} I_{\mathcal{C}} : & [\![1,\mathcal{W}]\!] \times [\![1,\mathcal{H}]\!] & \rightarrow & [0,1] \subset \mathbb{R} \\ & (x,y) & \mapsto & I_{\mathcal{C}}(x,y) = \mathcal{C}(s_{x,y}) \end{array} \qquad (12)$$

Given Definition 9, $\mathcal{C}(s_{x,y})$ can be computed for each sequence $s_{x,y}$ (*i.e.*, for each pixel (x,y)). Definition 10 gives the characteristic of the image representing the contribution of each pixel to a set of sequential patterns. Once this synthetic representation is obtained, the expert is provided with a valuable guide that allows him to identify in one viewing the kind of surfaces involved in the set of extracted patterns. For instance, if most of the highlighted pixels correspond to areas covered with fields, then the expert may browse the set of extracted patterns with phenomena such as "crop rotation" in mind.

6 Experiments

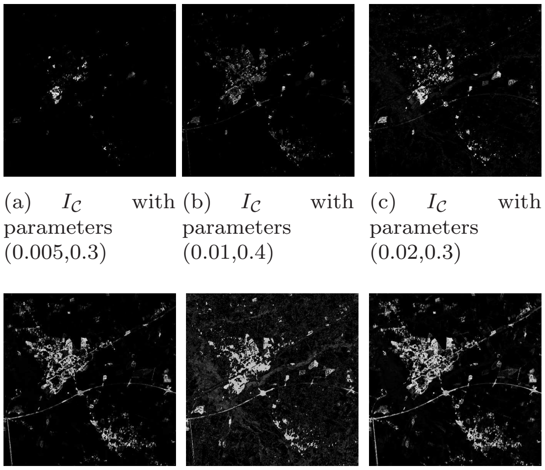
Our 35 images have a resolution of 202,500 pixels (450x450). Once preprocessed (as described in Section 4), each pixel takes values on four attributes (NIR, R, G, NDVI) and our data contain a total of 28 million values in the series. The values have been quantified according to Section 4, attribute by attribute. Table 2 links the quantified values to the sensed ones. By applying the method described in Section 5 to the SITS illustrated in Figure 4, we obtain a set of patterns that needs to be interpreted by the experts. In this section, we provide some results and their interpretation in the two first subsections. The last subsection is then devoted to the study of the maximum support and its efficiency regarding the time response.

Table 2: Correspondence between sensed values and quantified ones.

Quantified	Mean reflectance			
value	NIR	R	G	NDVI
1	60	5	7	0
2	93	8	11	0.28
3	111	10	13	0.38
4	121	13	16	0.45
5	128	16	18	0.49
6	132	18	20	0.54
7	137	22	21	0.57
8	144	26	22	0.61
9	152	28	25	0.64
10	162	29	27	0.67
11	174	30	30	0.70
12	184	32	33	0.72
13	194	35	35	0.75
14	203	39	37	0.78
15	214	42	40	0.81
16	227	47	46	0.84
17	242	56	56	0.87
18	263	69	78	0.90
19	293	93	138	0.92
20	343	157	160	0.96

6.1 Visualization of Data Mining Patterns on Image Time Series

Examples of different $I_{\mathcal{C}}$ computed for different $F^{\mathcal{S}}$ are displayed on Figure 7. These results aim at visually determining the content of the set of extracted patterns. Not surprisingly, the lower the minimum support, the lower the number of sequences supporting the extracted patterns (series of states of evolutions). Actually, with low supports, the extracted patterns get more and more specific and correspond to fewer pixels. Besides this obvious observation, Figure 7(a) shows that the pixels supporting the patterns with such low supports are mainly located in industrial zones. Figure 7(c) contains a little more sequences supporting the set of sequential patterns, mostly corresponding to urban areas and reclaimed land. As to Figure 7(e), it corresponds to urban areas and wet vegetation (swamps) in dark gray. This image provides us with another type of information: the constant gray-level of urban areas indicates that these areas are often supporting patterns all together.



(a) $I_{\mathcal{C}}$ with parameters (0.005,0.3)

(b) $I_{\mathcal{C}}$ with parameters (0.01,0.4)

(c) $I_{\mathcal{C}}$ with parameters (0.02,0.3)

(d) $I_{\mathcal{C}}$ with parameters (0.05,0.5)

(e) $I_{\mathcal{C}}$ with parameters (0.07,0.3)

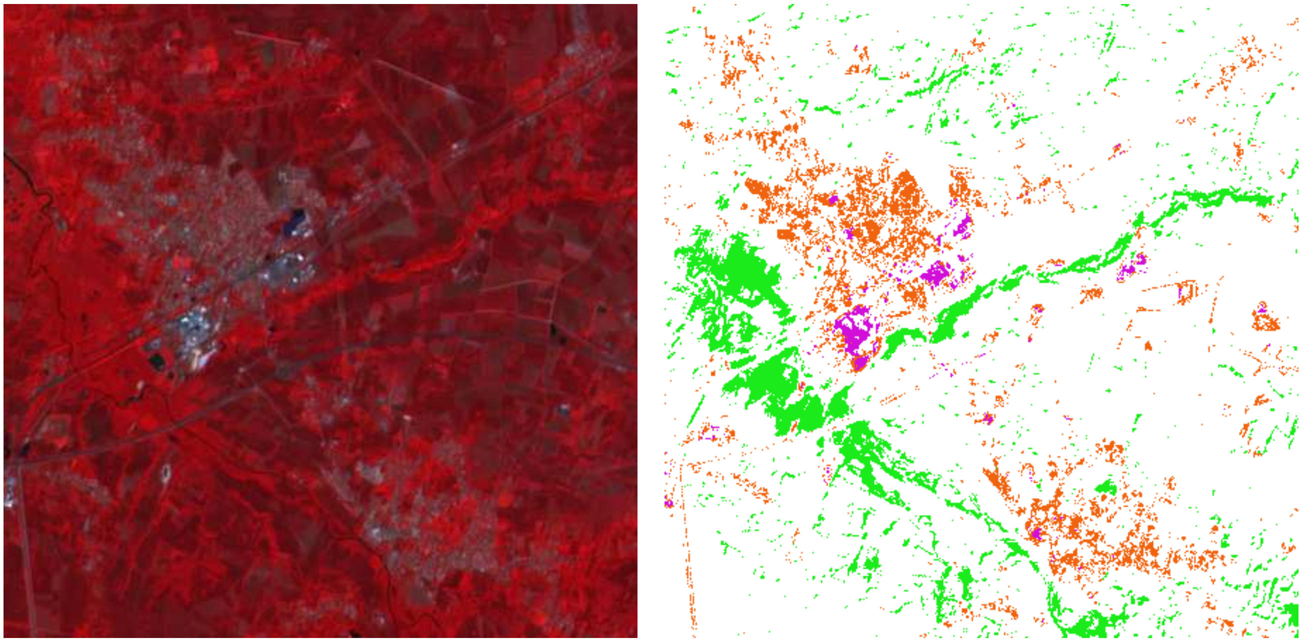
(f) $I_{\mathcal{C}}$ with parameters (0.07,0.5)

Figure 7: $I_{\mathcal{C}}$ images for different extractions with parameters (minimum support,maximum support). 256 gray levels (from black to white) are used in order to illustrate the contribution frequencies. White pixels correspond to high $\mathcal{C}(s_{x,y})$.

6.2 Exploring the Evolution Patterns

The images built on the principle described in the previous subsection (and illustrated in Figure 7) provide the experts with valuable intuitive information on the kind of patterns contained in the data mining result for a minimum and a maximum support. For each individual pattern in this result, we can retrieve the pixels whose series of values contain the pattern. These pixels may than be visualized (highlighted) as illustrated by Figure 8(b) (right). In this figure, each color corresponds to a pattern selected from our SITS. We also report an image from the series (left) for visualization purpose. This work is supported by the French Space Agency and these results where considered by experts in the domain. Here is the geographic explanation of these patterns.

The pattern <**(IR,1) (NDVI,20)**> is represented by the green dots in Figure 8(b). It cor-



(a) June 3 2006: image selected from the SITS

(b) Illustration of our result with three selected patterns (one color per pattern).

Figure 8: A sample of our results (right) compared for visualization purpose to one image from the series (left).

responds to **swamps** (wetlands) in the SITS and their **cyclic behavior**. During winter, swamps are almost covered with water, resulting in a low infra-red level (slice 1) since water does not reflect light a lot. During summer, these swamps are not covered with water any more and light is reflected by the vegetation. Due to its high chlorophyll concentration (due to high irrigation), vegetation in summer has a very high level in NDVI (slice 20).

The orange dots represent the pattern <**(R,17) (R,18 ; NDVI,3)**>. It corresponds to **urban areas that get denser** (the number of residences has grew). Actually, urban areas (residences) have a high response in the red band. The level at the beginning of the pattern (slice 17) is highly likely to be the sign of an urban area. The following level (slice 18) shows an urban densification (slices 17 and 18 are separated by a radiometric increase of nearly 15 %), confirmed by a low level of NDVI (corresponding to almost no vegetation). Extracting this kind of patterns is interesting because the urbanization behavior spreads over long time period, with different start time and end time. In this SITS, the building of new houses mostly starts in 1995 and was not finished in the end of the series. Moreover, the urbanization time can vary from one area to another. The extracted pattern concentrates all building behaviors, demonstrating the relevance of frequent patterns mining methods for the analysis of satellite image time series.

The pattern <**(NDVI,2) (G,20) (NDVI,1)**> is represented by the purple dots. This pattern corresponds to a **densification of industrial areas** (*e.g.* increase in the number of warehouses). In fact, industrial areas have high response in the green band and show very low values of NDVI. Furthermore, the significant decrease of NDVI shows that vegetation almost disappeared from these areas. Lastly, the maximum level of green is typical of flat roofs (*e.g.* corrugated iron) of industrial areas.

6.3 Efficiency of the Filters

The maximum support introduced in Subsection 5.1 has an important influence on the number of frequent items that are not allowed in successive combinations (and, therefore, on the number of sequential patterns that are evaluated during the process). In Figure 9 (left) we report the number of frequent items that can be combined into candidate sequential patterns. For instance, when the minimum support is 50 %, the number of frequent items ("Original") is 63, but with a maximum support of 75 %, the number of items

i having support falling between 50 % and 75 % drops to 37 (which corresponds to a reduction factor $h = 1.7$). We report in Figure 9 (right) the time responses with a maximum support of 80 %, 75 % and 70 %, and we compare it to an extraction with no maximum support ("Original"). We can observe that the influence of the reduction factor h is very important on the time response, as expected and described in Subsection 5.1 and Figure 6.

This principle is well suited for SITS mining, since it discards non-evolution patterns and it outperforms by at least one order of magnitude the process attempting to extract the whole set of frequent sequential patterns. For instance, in our experiments, we observed that with a minimum support of 50 % and no maximum support, the extraction needs several hours. With a minimum support of 10 %, we estimate that it should take several weeks. However, with a minimum support of 10 % and a maximum support of 50 %, the process completes within minutes.

7 Conclusion

Sequential patterns are well suited for evolution discovery and description. These patterns are extracted from series of values and they give the frequent successive values that are embedded in the series. The intrinsic properties of sequential patterns extraction algorithms fit the specific characteristics of satellite image time series: they are robust to meteorological noise, they tolerate irregular sampling of images and are able to extract behaviors of different length and with different start and end times. However, when it comes to extracting evolution patterns from SITS, the challenge is to filter out the very large number of non-evolution patterns, because of which the experts could be overwhelmed. In this paper, we have proposed i) a framework for evolution pattern mining from SITS, ii) a principle intended to filter out the non-evolution patterns and enable the mining process and iii) a visualization technique that makes it possible to locate areas of evolution in one viewing.

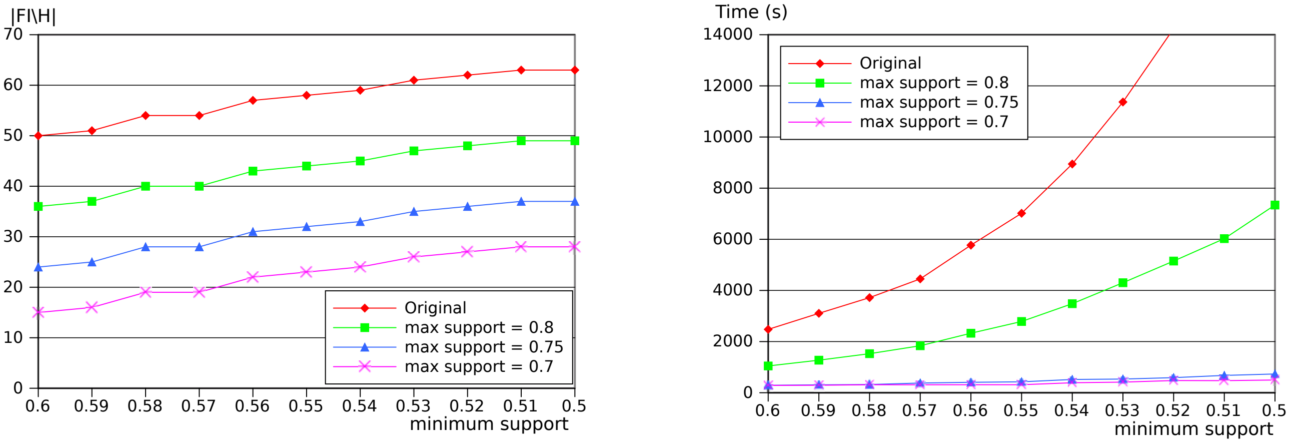
Experiments carried out on a particular dataset showed that FSPM methods allow us to extract repeated, shifted and distorted temporal behaviors. The flexibility of these methods makes it possible to capture complex behaviors from multi-source, noisy and irregularly sensed data. However, this flexibility could also have some drawbacks. In fact, this flexibility could lead to skipping minority states of evolution, even if they are characteristic of a class. Moreover, the acquisition dates of the images were not used as an information for these experiments. Finally, this kind of methods does not provide directly a whole classification of the sensed area, *i.e.*, there is not a class associated to each sensed area.

Considering these issues, we believe this work opens up a number of research directions. First, further experiments should be conducted in order to demonstrate the genericity of the proposed approach. Second, developed filters could include more sophisticated rules in order to integrate the knowledge of thematic experts. Third, when dealing for instance with agronomical classes, the date of the different states of evolution has to be used in order to discriminate one class from another. More work is still needed to integrate the use of the date of the images in a flexible way. Finally, inferring a clustering of the data from extracted patterns remains an important issue for both theoretical and applied research direction.

Finally, this work shows new interesting leads for sequential pattern extraction when the number of possible combinations does not allow to calculate the frequent sequences. This may have important consequences in applications where the data does not vary much and we want to find patterns with a frequency threshold that is not very high while remaining significant. This is the case, for instance, in Web Usage Mining, where the most frequent behaviors usually correspond to non informative patterns (*e.g.* "20% of users click on *homepage* followed by *location*"). In data mining, it is well known that depending on some characteristics of the data, one has to choose the approach that best corresponds. Our study shows a new approach to consider, depending on the data and their characteristics.

Acknowledgments

This work was supported by the CNES (French Space Agency) and Thales Alenia Space. The authors would like to thank M. Lanselle for his careful reading of the article.



(a) Number of items in $|FI\backslash H|$

(b) Time responses

Figure 9: Study of the number of items and time responses.

References

References

[1] Gouko, M., Ito, K.: An action generation model by using time series prediction and its application to robot navigation. International Journal of Neural Systems **19**(2) (2009) 105–113

[2] Möller-Levet, C.S., Yin, H.: Modeling and analysis of gene expression time-series based on co-expression. International Journal of Neural Systems **15**(4) (2005) 311–322

[3] Shi, L.D., Shi, Y.H., Gao, Y., Shang, L., Yang, Y.B.: XCSc: A novel approach to clustering with extended classifier system. International Journal of Neural Systems **21**(1) (2011) 79–93

[4] 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** (May 2004) 1565–1596

[5] Campbell, J.B.: Introduction to Remote Sensing. The Guilford Press (2007)

[6] Jensen, J.R.: Remote Sensing of the Environment: An Earth Resource Perspective. Prentice Hall (2007)

[7] Agrawal, R., Srikant, R.: Mining Sequential Patterns. In: Proceedings of the 11th International Conference on Data Engineering (ICDE'95). (1995) 3–14

[8] Masseglia, F., Cathala, F., Poncelet, P.: The PSP Approach for Mining Sequential Patterns. In: Proceedings of the 2nd European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD'98), Nantes, France (1998)

[9] Pei, J., Han, J., Mortazavi-asl, B., Pinto, H., Chen, Q., Dayal, U., Hsu, M.: Prefixspan: Mining sequential patterns efficiently by prefix-projected pattern growth. In: ICDE '01: Proceedings of the 17th International Conference on Data Engineering, Washington, DC, USA, IEEE Computer Society (2001) 215

[10] Julea, A., Méger, N., Trouvé, E.: Sequential patterns extraction in multitemporal satellite images. In: 17th European Conference on Machine Learning and the 10th European Conference on Principles and Practice of Knowledge Discovery in Databases (ECML / PKDD). (2006) 96–99

[11] Julea, A., Méger, N., Trouvé, E., Bolon, P.: On extracting evolutions from satellite image time series. In: IEEE International Geoscience and Remote Sensing Symposium (IGARSS). Volume 5. (2008) 228–231

[12] Masseglia, F., Poncelet, P., Teisseire, M., Marascu, A.: Web usage mining: extracting unexpected periods from web logs. Data Mining and Knowledge Discovery (DMKD) **16**(1) (2008) 39–65

[13] Bruzzone, L., Prieto, D.: Automatic analysis of the difference image for unsupervised change detection. IEEE Transactions on Geoscience and Remote Sensing **38**(3) (May 2000) 1171–1182

[14] Jensen, J.R.: Urban change detection mapping using landsat digital data. Cartography and Geographic Information Science **8**(21) (October 1981) 127–147

[15] Johnson, R., Kasischke, E.: Change vector analysis: a technique for the multispectral monitoring of land cover and condition. International Journal of Remote Sensing **19**(16) (1998) 411–426

[16] Foody, G.: Monitoring the magnitude of land-cover change around the southern limits of the Sahara. Photogrammetric Engineering and Remote Sensing **67**(7) (2001) 841–848

[17] Howarth, P., Piwowar, J., Millward, A.: Time-Series Analysis of Medium-Resolution, Multisensor Satellite Data for Identifying Landscape Change. Photogrammetric engineering and remote sensing **72**(6) (2006) 653–663

[18] Kennedy, R.E., Cohen, W.B., Schroeder, T.A.: Trajectory-based change detection for automated characterization of forest disturbance dynamics. Remote Sensing of Environment **110**(3) (2007) 370–386

[19] Andres, L., Salas, W., Skole, D.: Fourier analysis of multi-temporal AVHRR data applied to a land cover classification. International Journal of Remote Sensing **15**(5) (1994) 1115–1121

[20] Ketterlin, A., Gançarski, P.: Sequence similarity and multi-date image segmentation. In: Fourth Intl Workshop on the Analysis of Multitemporal Remote Sensing Images, Leuven, Belgique (july 2007)

[21] MacQueen, J.: Some methods for classification and analysis of multivariate observations. In: Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. (1967) 281–297

[22] Pei, J., Dong, G., Zou, W., Han, J.: Mining condensed frequent-pattern bases. Knowledge and Information Systems (KAIS) **6**(5) (2004) 570–594

[23] Soulet, A., Crémilleux, B.: Adequate condensed representations of patterns. In: Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML / PKDD), Springer-Verlag (2008) 20–21

[24] Jeudy, B., Boulicaut, J.: Using condensed representations for interactive association rule mining. In: Proceedings of the 6th European Conference on Principles of Data Mining and Knowledge Discovery. (2002) 225–236