TRAJECTORY OF DYNAMIC CLUSTERS IN IMAGE TIME-SERIES

P. HEAS[1,2]
[1]*IRIT* / [2]*DLR*
[1] *2 rue Camichel, 31071 Toulouse, France*
E-mail: Patrick.Heas@enseeiht.fr

M. DATCU[2], A. GIROS[3]
[2]*German Aerospace Center, DLR Oberpfaffenhofen, Germany*
[3]*CNES, Toulouse, France*
E-mail: Mihai.Datcu@dlr.de; Alain.Giros@cnes.fr

In this paper, a dynamic scene understanding concept is proposed and applied on multispectral image time-series. Information mining enables the exploration and discovery of spatio temporal patterns localized in given spatio temporal windows. With this in mind, a hierarchical information representation which comprises several processing steps is developed. The features are extracted locally in a time window. Several classifications are then performed for each time window according to a Gaussian mixture model constrained by a minimum description length criterion. The non-parametric clustering algorithm is presented and evaluated. Then, the cluster dynamic is investigated in order to enable a graph representation displaying flow of feature points. The analyzis is also performed in the space of the image time-series. Lastly, the concept is applied on remotely sensed data and a couple of pattern behaviors are presented.

1. Introduction

1.1. *Times series of satellite images*

Time-series, or more generally, temporal sequences, appear naturally in a variety of different domains, from engineering to scientific studies, finance and medicine. Satellite image sequences are different from usually considered image time-series. They contain useful information in various space-scales including objects at pixel scale or large structures spread on hundreds of pixels. Moreover, the acquisition depends on various constraints which impose an irregular sampling rate and a long acquisition period to build a consistent dataset. Consequently, the temporal phenomenon considered

can be at various time-scales which include events occurring on a range of years, months, shorter events such as occlusion by clouds, wind effects on water, etc, or even evanescent phenomena.
Accordingly, we should consider the above properties in the architecture of a dynamic scene understanding concept, able to mine patterns in space and time at various scales, thus with a global or precise spatio-temporal location and to discover the interactions and causalities between them.

1.2. *Information mining by analyzing the cluster dynamics*

To achieve the goal of dynamic scene understanding and information mining, these spatio-temporal patterns can be modeled by analyzing the cluster changes in their feature spaces.
In this paper, a minimum description length based clustering algorithm is applied to model, in consecutive times and according to a Gaussian mixture assumption, features extracted from temporal windows. New methodes, based on cross entropy, are developed to quantify the changes of the clusters and thus, to formalize them as cluster trajectories.

2. Investigating the dynamics of clusters

2.1. *Multitemporal clustering*

An important issue for the understanding of spatio-temporal scenes is the process of finding "good" models for the hierarchical information representation along its transmission through the architecture. This standard concept previously developed in *I2M*[5] [2] for the mining of static scenes was adopted and redesigned for multitemporal information mining[4] Exploration is driven by the interaction of a user, in a Bayesian framework. The learning process updates posterior maps A threshold of these maps produces the desired classifications.

2.2. *Time-localized clustering*

So far in this paper, the considered classifications were multitemporal. Indeed, information from the whole time-series was processed with no defined time location t_i. Mining in a time-window over the time-series enables time dependent classifications $f(x_{t_i})$ of localized spatio-temporal patterns where the x_{t_i} are the features extracted in a given time-window related to time t_i. The biggest time sample index is denoted by N. A global time-localized clustering $f(\cup_{i=1}^{N}\{x_{t_i}\})$ produced by the union of all the features extracted

from each windowed image time-series $\cup_{i=1}^{N}\{x_{t_i}\}$, leads to a single classification of all the time-localized spatio-temporal patterns present in the different windows. Thus, each time-window in the time-series possesses a corresponding classification.

2.3. *Analyzes of the dynamics of the feature space*

Nevertheless, in a time-localized clustering, the information about the causalities between the windowed time-series is neglected. To obtain this complementary information, an investigation of the dynamics of the feature space is necessary. The temporal behavior of the features grouped in clusters is a complex matter. There are various sorts of exchanges of features between the clusters in time. Mainly, there are stable behaviors characterized by cluster evolutions in which the cluster features are distinguished from the others. There also exist evolutions such as the splitting or the merging of clusters in which there are relatively to some features, exchanges of populations between the concerned clusters. In order to detect such phenomena, a classification $f(x_{t_i})$ of each windowed time-series has to be performed. The crucial problem of estimating the number of clusters for each time remains. The causalities between the time dependent classifications has also to be modeled.
Solving these two problems will enable us to build a graph of flowof feature points specific to the image time-series. The features specific to each node are the dynamic cluster attributes around the given node (centroid trajectory values and their associated covariance evolutions in the case of a evolving Gaussian mixture), the flow of features points between these nodes and the splitting/merging phenomena.

2.4. *Proposal of solutions for dynamic cluster modeling*

2.4.1. *Minimum description length (MDL) principle for Gaussian mixture modeling*

Let us make the assumption that the data is distributed with a d dimensional Gaussian mixture (GM) distribution defined for K Gaussian as

$$f_K(x) = \sum_{k=1}^{k=K} \pi_k \mathcal{N}_d(M_k, A_k) \ , \quad \sum_{k=1}^{k=K} \pi_k = 1 \tag{1}$$

where $\mathcal{N}(M_k, A_k)$ is a normal distribution of mean M_k and covariance A_k, and π_k a weighting fuction . In order to perform classifications without any

constraint neither on the number of Gaussian present in the mixture nor on their parameters, we need to define a criterion to select the best model among all the possible GM configurations. The MDL principle gives us a solution to the problem.

The MDL principle

Let us present the MDL principle by introducing a few considerations : First, there is a 1-1 correspondence between probability distributions and code length functions. Given data $x^n = \{x_1, ..., x_n\}, \forall x^n, \forall n$

$$L(x^n) = -log\big(P(x^n)\big) \tag{2}$$

where $L(x^n)$ denotes the length (in bits) needed to describe x^n and P a probability distribution. Then, let M be a probabilistic model, i.e. a finite set of distributions : $M = \{P(.|\theta_1), ..., P(.|\theta_M)\}$. P_M is a universal model for M, if

$$-log\big(P_M(x^n)\big) \leqslant -log\big(P(x^n|\theta)\big) + K \tag{3}$$

where K is a constant independent of n.

In the perspective of equation 2, a universal model called the "2-part MDL code" is derived. It is the length of coding x^n by first coding $\widehat{\theta}(x^n)$, then coding x^n with the help of $\widehat{\theta}(x^n)$. It is defined by

$$L_{2P}(x^n) = -log\{W(\widehat{\theta}(x^n))\} - log\{P(x^n|\widehat{\theta}(x^n))\} \tag{4}$$

where W is a prior over M and $\widehat{\theta}$ is an estimate corresponding to the function minimum. The second term can be seen as the log-likelihood function. Obviously, the code length for coding the prior W depends on the prior function over M that we are considering. The 2-part MDL code is not an optimal universal model and there are some equivalences with the Bayesian inference[6]. However, its computation is easier than the calculation based on a Bayesian framework used for instance in the Autoclass algorithm[3]. The MDL principle holds for all universal models. It states that the best model among a collection of tentatively suggested ones is the one that gives the smallest stochastic complexity to the given data. The stochastic complexity is represented by the chosen universal model.

Application to Gaussian mixture modeling

On the basis of the 2-part MDL code, a criterion is derived, corresponding to the length of coding the data assuming a Gaussian mixture distribution. The estimated model (in the sense of the MDL principle) is used to

perform a non parametric clustering. A simplified model, assuming each data component independent from the others and all clusters with constant variances, was previously developed in[1].
The first part composing the cost function is the length of code used for coding the model itself :
Each Gaussian of the mixture must have its parameters coded and the code length for the K Gaussian is

$$-log\{W(\widehat{\theta}(x^n))\} = K(L_1 + L_2) \tag{5}$$

where L_1 and L_2 are respectively the number of bits necessary to code a mean vector and a covariance matrix. A remark is that in the clustering perspective, we do not intend to favor either independence or correlation between the data components. In other words, the prior function is equal for these two configurations and we do not intend to code the covariance matrix optimally to reduce the code length L_2. Equally, no optimal coding will be performed on L_1.
The second part composing the criterion is the number of bits necessary to code the data using the GM parameters : We decompose this coding in two parts: a) the coding of the index of the n data points to their respective Gaussian and 2) the coding of the indexed data points according to their respective Gaussian probability distributions. The number of bits needed to encode all the index is given by

$$-\sum_{k=1}^{K} n_k log_2 \frac{n_k}{n} \tag{6}$$

where n_k is the number of data points assigned to the class k and n the sample size. The number of bits needed to encode the indexed data points is

$$\sum_{k=1}^{K} \sum_{x_i \in G_k} \Big(-\frac{1}{2} log_2 \frac{|A_k^{-1}|}{2\pi^d} + \frac{1}{ln(2)}{}^t(x_i - M_k) A_k^{-1} (x_i - M_k) \Big) \tag{7}$$

where G_k is a the Gaussian k with its mean vector M_k and its covariance matrix A_k.
Thus, adding the two parts we obtain the cost of encoding x^n using the GM model

$$L_{2P}(x^n) = K(L_1 + L_2) \tag{8}$$
$$-\sum_{k=1}^{K} \Big[n_k log_2 \frac{n_k}{n} + \sum_{x_i \in G_k} \Big(-\frac{1}{2} log_2 \frac{|A_k^{-1}|}{2\pi^d} + \frac{1}{ln(2)}{}^t(x_i - M_k) A_k^{-1} (x_i - M_k) \Big) \Big]$$

Optimization

Our goal is to minimize $L_{2P}(x^n)$, i.e., determine K and $\{(M_k, A_k)\}$. Enumerating all configurations and evaluating the cost function is not feasible. Instead, we use an optimization algorithm which evaluates the changes of the code length between two configurations rather than the code length itself.

Before introducing the general algorithm, let us present the change of the code length if a given Gaussian $G_{k'}$ is removed. One can proove that this quantity is defined by the expression

$$\begin{aligned}\Delta_{G_{k'}}(L_{2P}(x^n)) = & -(L1+L2) + n_{k'} log_2 \frac{n_{k'}}{n} \\ & - \sum_{l=1, l\neq k'}^{K} \left((n_l + n_{lk'}) log_2 \frac{(n_l + n_{lk'})}{n} - n_l log_2 \frac{n_l}{n} \right) \\ & + \sum_{x_i \in G_{k'}} log_2 \frac{p_{k'}(x_i)}{p_{k'k'}(x_i)} + \sum_{l=1, l\neq k'}^{K} \sum_{x_i \in G_l} log_2 \frac{p_l(x_i)}{p_{lk'}(x_i)} \end{aligned} \quad (9)$$

where n_l and $n_{lk'}$ are respectively the number of data points assigned to the class l before and after removal of $G_{k'}$, and where p_l and $p_{lk'}$ are respectively the probability distribution of G_l before and after removal of $G_{k'}$. $p_{k'k'}(x_i)$ is the probability distribution of the most probable Gaussian which is assigned to x_i after removal of $G_{k'}$.

The optimization algorithm is composed by the following steps :

1-Initialization : A GM with a high number of Gaussian K and given parameters $\{(M_k, A_k)\}$ is produced. There are many strategies to perform such an initialization.

2-Adaptation : An EM-type algorithm is used to estimate the parameters $\{(M_k, A_k)\}$.

3-Selection : Evaluation of the changes in $L_{2P}(x^n)$ while we simulate the removal of each G_k for $k \in [1, K]$. If there are more than one decrease of $L_{2P}(x^n)$,the G_k corresponding to the maximum decrease is removed.

4-Convergence : If the selection step has not removed any G_k and the adaptation step has reached convergence then 'exit', otherwise 'go to step 2'.

The initialization is done by spreading randomly K_{init} clusters according to a Gaussian distribution of mean and variance learnt from each data component. With such an initialization and performing the algorithm on simulated Gaussian mixtures with a sufficiently large sample size, the true

number and parameters of the GM is recovered, even for very intricate GM. Of course training on an insufficient number of samples and considering too 'close' Gaussians in the mixture does not permit to find the true parameters. Thus, the MDL based Gaussian mixture clustering algorithm, which is able for sufficiently large sample size, to estimate the true number of clusters and their parameters.
However, in spite of an initialization of the cluster center values with the previous classification, the complexity of the flow of feature points does not enable clear trajectories identification.

2.4.2. *Modeling a Gaussian mixture evolution*

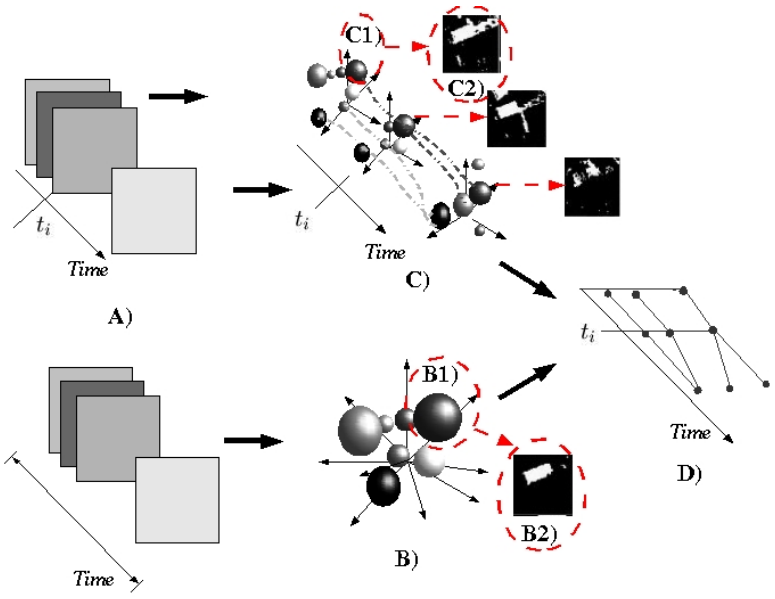


Figure 1. Dynamic clusters modeling; A: Data, B:Multitemporal clustering, C:Time-localized clustering, D: Graph characterizing the dynamic clusters.

Unfortunately, the traceability of the clusters attributes which are in the case of a Gaussian mixture modeling resumed to the centroid trajectories and the covariance evolutions, is not an easy task. The irregular sampling and the difficulty of the calibration between the image time samples forbid a direct modeling of the flow of feature points and in particular a direct Gaussian mixture evolution modeling. To face these causality problems, which

are necessary for the construction of the graph of flow of feature points, a multitemporal classification over the complete image time-series exhibiting spatio-temporal patterns is required. Such a classification has been described previously in section 2.1. Actually, possessing a multitemporal classification $f(x_{t_1},...,x_{t_N})$ of spatio-temporal patterns and several time-localized ones $\{f(x_{t_i}); i \in [1,N]\}$, and assuming a model to measure the distance between the multitemporal clusters and the time-localized clusters, it is possible to trace the cluster evolutions. Hence, by assigning multitemporal classes to each time-localized class, the graph of flow of feature points in which each node contains the center and covariance values, the quantity of feature points exchanged and the splitting/merging information can be built. The concept is schematized in figure 1.

The problem of measuring the distance between multitemporal clusters $C^{j_1}_{t_1,...,t_N} \in f(x_{t_1},...,x_{t_N})$ and time-localized clusters $C^{j_2}_{t_i} \in f(x_{t_i})$ can be solved by estimating the probability distribution function (p.d.f.) of the clusters in order to evaluate their cross entropy. j_1 and j_2 represents respectively the index of a Gaussians of the mixture in time t_i and the index of a Gaussian of the mixture for the multitemporal classification. Obviously, the multitemporal features considered here must be commensurable with the features used for the time-localized classifications thus, the multitemporal clusters must be projected in the time-localized feature space to enable a coherent cross entropy measurement. At time t_i, each multitemporal clusters $C^{j_1}_{t_1,...,t_N}$ for $j_1 \in [1,K]$ is assigned to one of the time-localized clusters $\{C^{j_2}_{t_i}; j_2 \in [1,K_{t_i}]\}$ according to a maximum cross entropy criterion formalized as

$$H\big(p_{C^{j_2}_{t_i}}(x) || p_{P_{t_i}\{C^{j_1}_{t_1,...,t_N}\}}(x)\big),$$

where K and K_{t_i} are respectively the number of multitemporal clusters and the number of time-localized clusters at time t_i, where $P_{t_i}\{.\}$ is a projection operator projecting the multitemporal feature space in the subspace embeding the $C^{j_2}_{t_i}(x)$ and where $p_{\{.\}}(x)$ are d dimensional Gaussian p.d.f. In order to simplify the notations, let us denote respectively by E and F a clusters $C^{j_2}_{t_i}$ and a cluster $P_{t_i}\{C^{j_1}_{t_1,...,t_N}\}$. The cross entropy measure is defined between the two d dimensional Gaussian p.d.f $p_E(x)$ and $p_F(x)$ by the expression :

$$H(p_E(x) \;||\; p_F(x)) = \frac{1}{2}\int_{\mathbf{R}^d} p_E(x).\Big(log\frac{|A_E^{-1}|}{|A_F^{-1}|} + \Big[{}^t(x-M_F)A_F^{-1}(x-M_F) - {}^t(x-M_E)A_E^{-1}(x-M_E)\Big]\Big)dx \quad (10)$$

where A_E and A_F are the associated covariance matrixes and M_E and M_F their respective mean vectors.

3. Results

The experiments were performed on a database subset which consists of 10 multispectral SPOT remotely sensed images of 100x100 pixels of a rural scene in Romania. The time-localized features were simply the radiometric values in a temporal window size 1 (for each image time sample). In this case, the features which are the original images are displayed in the middle of figure 2. A multitemporal and 10 time-localized clusterings were respectively produced on the space constituted by the union of the 10 time-localized features and by each of them separately. We obtained 9 Gaussians for the multitemporal classification and cluster numbers fluctuating from 6 to 9 for the time-localized classifications. At first sight, it is likely that the number of Gaussians representing the 9 class evolutions was underestimated. Then, the Gaussian mixture evolution was modeled using the cross entropy measurements. Thus, the graph was infered. In figure 2, a graph displaying some of the feature information is drawn. The horizontal axis represents time (in correspondence with the original images above); the vertical axis represents the red radiometric value. Each node, on which the assigned multitemporal classes are written and the variances represented (vertical black lines), is linked to the past and future nodes. Thus, merging and splitting information in the red spectral band is available. In parallel, the above table of images in figure 2 presents the spatial class evolutions. The first column displays the 9 multitemporal classes indexed upward from "0" to "8". The next columns display these 9 class evolutions (time is increasing from left to right). Each image displays in principle, a single time-localized class including at least one multitemporal class. Two important remarks can be made. First, as expected, the number of multitemporal classes was underestimated and consequently in some images, we observe "interclass-splitting" phenomena : two time-localized classes are regrouped in a single multitemporal class (the other class is colored in gray in figure 2). Second, one can of course notice that at a given time, some other connected components are sharing the same mulitemporal class.
In order to interprete the results, one can intersect some selected information directly in the feature-graph and in the spatial class evolutions in order to characterize some typical behaviors with a time and a spatial address. For instance, one can notice that in "time sample 5", the "multitemporal

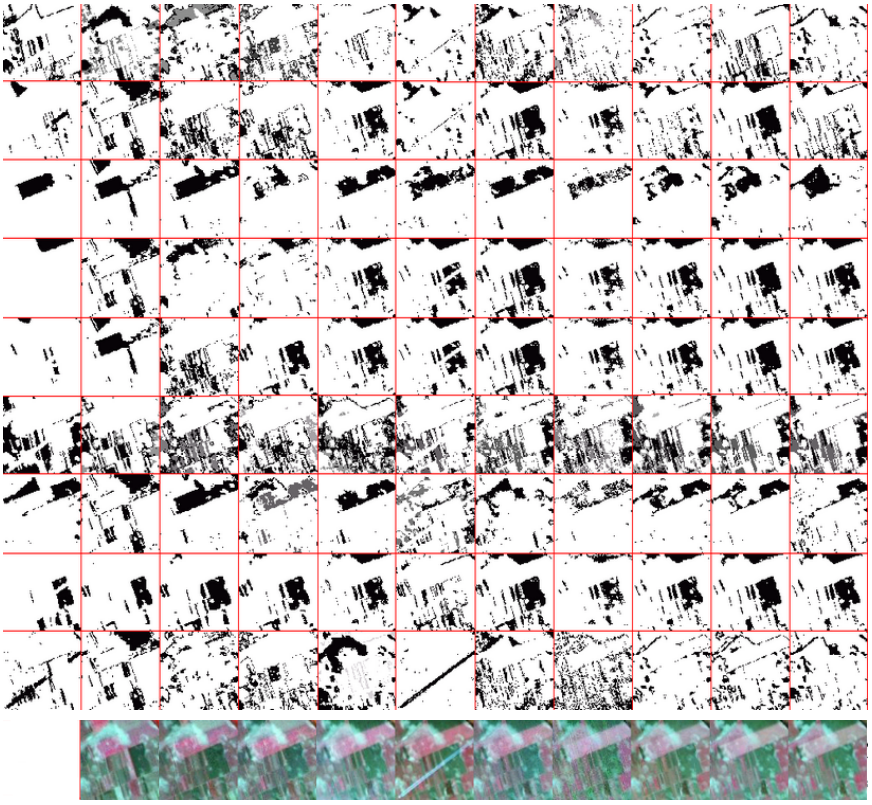
class 0" (in blue) is the only present class in a node. But in "time sample 4 and 6", "class 0" shared the same node as the multitemporal class 8. We observe here a splitting and a merging phenomena, along with a radiometric jump corresponding to a plane apparition. Indeed, this is obvious in the original images and in the spatial evolution of the multitemporal classes 0 and 8. This typical behavior can be used to identify other similar phenomena.

4. Conclusion

This work is an attempt to solve the complex problem of spatio-temporal patterns recognition in image times series. A flow of feature points modeling of the dynamic feature space is proposed to extract information related to the trajectory of clusters in a time-window. The modeling is decomposed into two processing levels :
1- The Gaussian mixture assumption is used to perform a multitemporal and several time-localized non-parametric clustering according to a MDL criterion.
2- The Gaussian mixture evolution is modeled on cross-entropy measurements. It leads to the information decomposition detailed above.
A couple of queries investigating into the graph are made and some pattern behaviors are pointed out. The concept presented is generic and can be extended to all kind of spatio-temporal features.

References

1. Bischof H, Leonardis A., and Selb. A.,*MDL Principle for Robust Vector Quantization,* Pattern Analysis and Applications, 2:5972, 1999.
2. Bruzzone L., Fernàndez Prieto D., *A Combined Supervised and Unsupervised Approach to Classification of Multitemporal Remote Sensing Images*, Proc. of the IEEE 2000 Int. Geoscience and Remote Sensing Symposium (IGARSS'2000), Honolulu, Hawaii, 24-28 July, 2000, Vol. I, pp. 162- 164.
3. Cheeseman P. *AutoClass: A Bayesian classification system.* Proceedings of the Fifth International Conference on Machine Learning, Ann Arbor, MI., June 12-14 1988. Morgan Kaufmann Publishers, San Francisco, 1988, pp. 54-64.
4. Heas P, Datcu M., Giros A. *Image time-series mining for dynamic scene understanding*, Workshop KDD2002 on temporal data mining, Canada, 2002.
5. Schroeder M, Rehrauer H., Seidel K. and Datcu M. *Interactive Learning and Probabilistic Retrieval in Remote Sensing Image Archives*, IEEE Trans. on Geoscience and Remote Sensing, pp. 2288–2298, 2000.
6. Rissanen J. *Lectures on Statistical Modeling Theory*, Helsinki Institute for Information Technology, Finland, 2002.



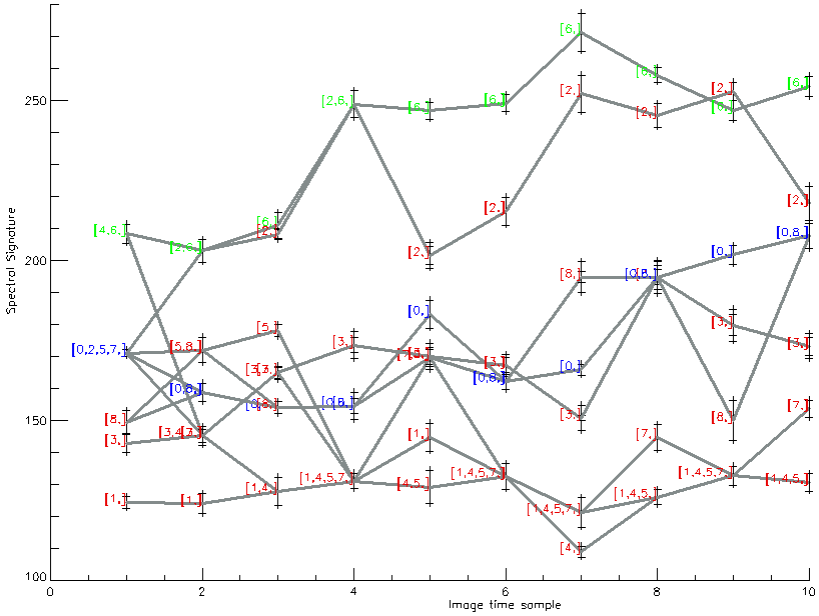


Figure 2. Class spatial evolutions in correspondence with the original time-series and the graph characterizing the dynamic clusters.