Constrained Distance Based Clustering for Satellite Image Time-Series

Thomas Lampert, Baptiste Lafabregue, Thi-Bich-Hanh Dao, Nicolas Serrette, Christel Vrain, Pierre Gançarski

▶ **To cite this version:**

HAL Id: hal-02325033
https://hal.science/hal-02325033

Submitted on 25 Oct 2019

Constrained Distance Based Clustering for Satellite Image Time-Series

Thomas Lampert[*], Baptiste Lafabregue[*], Thi-Bich-Hanh Dao[†], Nicolas Serrette[†],

Christel Vrain[†],

and Pierre Gançarski[*]

[*]ICube, University of Strasbourg, Strasbourg, France [†]LIFO, University of Orléans, Orléans, France

Abstract

The advent of high-resolution instruments for time-series sampling poses added complexity for the formal definition of thematic classes in the remote sensing domain—required by supervised methods— while unsupervised methods ignore expert knowledge and intuition. Constrained clustering is becoming an increasingly popular approach in data mining because it offers a solution to these problems, however, its application in remote sensing is relatively unknown. This article addresses this divide by adapting publicly available constrained clustering implementations to use the dynamic time warping (DTW) dissimilarity measure, which is sometimes used for time-series analysis. A comparative study is presented, in which their performance is evaluated (using both DTW and Euclidean distances). It is found that adding constraints to the clustering problem results in an increase in accuracy when compared to unconstrained clustering. The output of such algorithms are homogeneous in spatially defined regions. Declarative approaches and k-Means based algorithms are simple to apply, requiring little or no choice of parameter values. Spectral methods, however, require careful tuning, which is unrealistic in a semi-supervised setting, although they offer the highest accuracy. These conclusions were drawn from two applications: crop clustering using 11 multi-spectral Landsat images non-uniformly sampled over a period of eight months in 2007; and tree-cut detection using 10 NDVI Sentinel-2 images non-uniformly sampled between 2016 and 2018.

Index Terms

Satellite image time series, land cover, classification, constrained clustering, semi-supervised clustering, background knowledge, partition clustering.

I. INTRODUCTION

Remote sensing problems are often characterised by large volumes of data and a lack of reference data [1, 2, 3, 4]. Satellite Image Time Series (SITS) are becoming more readily available with the introduction of satellite constellations that can capture the same location with increasingly regular intervals [3].

Spatio-temporal classification opens up new possibilities in remote-sensing—to monitor land use and its evolution [4, 5, 6, 7, 8, 9, 10]. Nevertheless, it complicates reference data collection because of the complexity of thematic classes and their lack of formal specification [3, 4]. As such, unsupervised clustering is often used [4, 9, 11, 12, 10], which offers a solution based upon the data alone. These approaches, however, ignore expert knowledge and intuition (i.e. potential thematic classes), and do not allow an expert to propose modifications to the clustering.

We define "intuition" as non-formalised knowledge that may be partial and may be imperfect. In general, an expert has an objective and an intuition of what they are looking for when analysing spatio-temporal data. In a supervised setting it is necessary to express this intuition through examples that are both representative in number (i.e. sufficient examples to fully represent each class) and quality (i.e. examples sufficiently represent a class' variance) [3]. This is unrealistic in satellite image time-series analysis as this type of data is relatively recent in remote sensing and, as such, an expert may only have a vague idea of the number of classes and their nature. The work contained herein, nevertheless, allows them to improve the results of unsupervised clustering in applications that are still to be fully formalised (i.e. they have a lack of nomenclature and formal description of classes).

Constrained clustering (alias semi-supervised clustering) is the process of introducing background knowledge (alias side information) to guide a clustering algorithm [13, 14]. The background knowledge takes the form of constraints that supplement the information derived from the data—through a distance metric—for a (generally small) subset of the data. The solution of a constrained algorithm strives to balance the information derived from the data with that from the constraints.

This article studies semi-supervised approaches due to their potential to ease the burden of reference data collection required for supervised classification, which is not possible when dealing

with 'big data' and, as discussed above, difficult when dealing with time-series. As such these approaches offer a tool for remote sensing researchers that is distinctly lacking in the literature with respect to time-series data.

This is addressed two-fold by presenting an overview of constrained clustering methods and how they can be adapted to this use case, and a comparative study of their performance on time-series datasets. The comparative study is achieved by selecting the most representative algorithms from the literature and adapting them to be suitable for time-series clustering where necessary. These are compared for crop classification which greatly benefits from the analysis of spatio-temporal data [5, 15, 16, 17, 18, 19, 20, 21, 22] and tree-cut detection [23, 24]. As such, this article offers insight into the different formulations of constrained clustering algorithms, how they can be adapted for use in satellite image time-series, and insight into which algorithms are suitable for such applications.

In the remainder of this article Section II presents some background information on clustering, user-constraints, time-series clustering, and presents constrained clustering algorithms that are prominent in the field. Section III describes the modification of publicly available implementations for use in remote-sensing time-series clustering and the methodology used during experimentation. Section IV presents a comparative study of the algorithms for agricultural crop classification and their application to tree-cut detection. Section V discusses these results and the limitations of existing approaches when applied to time-series data. And conclusions are drawn in Section VI.

II. BACKGROUND

A. Cluster Analysis

Let $\mathcal{O}$ be a set of instances (data points) $\{o_1, \ldots, o_n\}$ and $d(o_i, o_j)$ a dissimilarity (or a similarity) measure between any two instances o_i and o_j. The similarity or dissimilarity between instances can be computed from their features or given by a similarity graph. Partition clustering involves finding a partition of $\mathcal{O}$ into K non-empty and disjoint groups called clusters, $C_1, \ldots, C_K$, such that instances in the same cluster are very similar and instances in different clusters are different. Cluster homogeneity is usually formalised by a optimisation criterion, and clustering aims at finding a partition that optimises the given objective. Different optimisation criteria exist for distance-based clustering, the most popular being [25]:

 (a) minimising the maximal diameter of the clusters,

 (b) minimising the maximal radius of the clusters,

   (c) minimising the sum of stars (the sum of the distances between each object and its cluster centroid),

   (d) minimising the within-cluster sum of dissimilarities,

   (e) minimising the within-cluster sum of squares.

   (f) maximising the minimal split between clusters,

Criteria (a)–(e) bias clusters to contain objects that are similar to each other, and criteria (f) biases different clusters to represent distinct objects.

B. User Constraints

In practice, a user may have some requirements for, or prior knowledge about, the final solution. For instance, the user can have some information on the label of a subset of objects [26]. Because of the inherent complexity of clustering optimisation criteria, classic algorithms always find a local optimum. Several optima may exist, some of which may be closer to the user requirement. It is therefore important to integrate prior knowledge into the clustering process and several studies have demonstrated the importance of this kind of domain knowledge in data mining processes [27]. Prior knowledge is expressed by user constraints to be satisfied by the clustering solution. The subject of these user constraints can be the instances or the clusters [28].

Instance-level constraints are the most widely used type of constraint and were first introduced by Wagstaff and Cardie [26]. Two kinds of instance-level constraints exist: *must-link* (ML) and *cannot-link* (CL). An ML constraint between two instances o_i and o_j states that they must be in the same cluster: $\forall k \in \{1, \ldots, K\}, o_i \in C_k \Leftrightarrow o_j \in C_k$. A CL constraint on two instances o_i and o_j states that they cannot be in the same cluster: $\forall k \in \{1, \ldots, K\}, \neg(o_i \in C_k \wedge o_j \in C_k)$.

In *semi-supervised clustering*, a small amount of labeled data is available to aid the clustering process. Instance-level constraints can be inferred from class labels: if two objects have the same label then they are linked by an ML constraint, otherwise by a CL constraint. Supervision by instance-level constraints is however more general and more realistic than class labels. Using knowledge, even when class labels may be unknown, a user can specify whether pairs of points belong to the same cluster or not [13]. Semi-supervised clustering is therefore a transductive operation because its objective is to define the clusters to explain the processed data and possibly to label the objects not initially labeled. During semi-supervised clustering, labeled objects and unlabeled objects are used to construct a clustering function. Thus, the objective is to use unlabeled objects to better understand the data space.
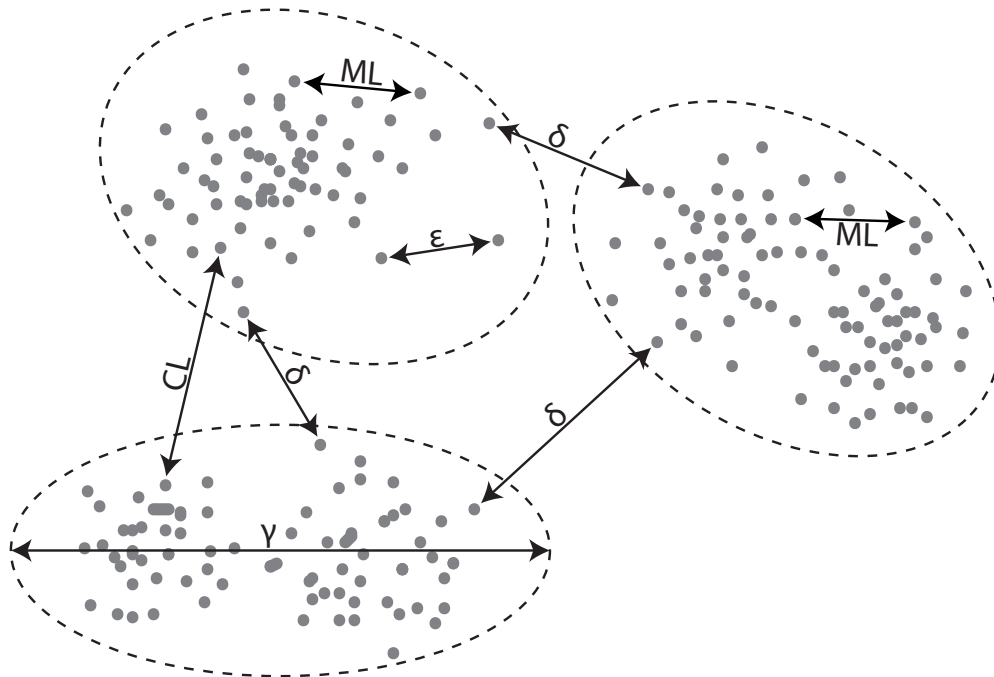
Figure 1. Examples of ML, CL, split (minimum separation) δ, maximum diameter γ, and ϵ (each object o_i has in its neighbourhood of radius ϵ at least one other object in the same cluster) constraints. A must-link (ML) constraint between two instances states that they must be in the same cluster and a cannot-link (CL) constraint on two instances states that they cannot be in the same cluster.

Cluster-level constraints define requirements on the clusters, for example:

- the number of clusters K;
- their absolute or relative maximal or minimal size – allows the user to control the balance of cluster cardinalities that an image should be clustered into;
- their maximum diameter, i.e. clusters must have a diameter of at most γ – allows the user to control the amount of similarity between objects in the same cluster;
- their split, i.e. clusters must be separated by at least δ – allows the user to control the level of difference that separates objects into different clusters;
- the ϵ-constraint [29] demands that each object o_i has in its neighbourhood of radius ϵ at least one other object in the same cluster – allows the user to remove small, sparse clusters that would cause noise in the clustered image.

See Figure 1 for an example of these constraints.

Mechanisms to integrate these constraints into the clustering process can be categorised into

three different approaches:

- enforcing constraints by guiding clustering algorithms during their process or by modifying the objective function;
- learning the distance function using metric learning;
- declarative and generative methods.

The requirement to constrain the number of clusters K is inherent in many algorithms (e.g. COP-KMeans, MIP-KMeans, Spec, and CCSR). In remote sensing problems, however, it is usually unknown. This is often mitigated by testing a range of K, however, some algorithms, e.g. SAMARAH and CPClustering, allow for K to be determined automatically, removing this burden from the user (although constraining it to a specific value is possible).

By far the easiest constraints to be interpreted by a user are must-link and cannot-link constraints because they do not require in-depth knowledge of the underlying clustering process and feature space. As such, this work will focus on algorithms that explicitly model these constraints. These are the most intuitive in remote sensing, as they are simply the expressions of whether a pair of objects (pixels, segments, time-series, etc.) should belong to the same cluster or not and can therefore be captured directly from an image.

C. Time-Series Clustering

Time-series increase the complexity of clustering due to the properties of the data. Almost all clustering algorithms use a distance function based upon the norm of two vectors L_p (Manhattan, L_1; Euclidean, L_2; and Maximum, L_∞). This implies a fixed mapping between points in two time-series and as such, norm based distances are sensitive to noise, misalignment in time (however small) [30], and are unable to correct for sub-sequence, i.e. non-linear, time shifts [31]. This may be desirable for the intended application, however, Dynamic Time Warping (DTW) [32, 33] finds an optimal alignment between two time series by non-linearly warping them. As such, it overcomes these, potentially undesirable, properties of norm based distances when applied to time-series. Furthermore, certain types of clustering algorithms, for example k-Means, calculate centroids during their optimisation, which is not a trivial task in the case of time-series due to the misalignments discussed previously. The DTW Barycenter Averaging (DBA) algorithm [34] overcomes this limitation by iteratively refining an initial estimate of the average sequence (usually taken to be a random sample of the time-series being averaged), in order to minimise its squared DTW measure to the sequences being averaged. As such, classical

constrained clustering implementations require modification to use DTW and DBA averaging (if required) before application to time-series. Other considerations when working with time-series are the dimensionality, which can be very large, meaning a sparse sampling of the input space.

A number of works give an in-depth background on clustering time-series [30, 35, 36, 37, 38, 39, 40]. Keogh and Lin [41] define two categories of time-series clustering:

- Whole Clustering: "The notion of clustering here is similar to that of conventional clustering of discrete objects. Given a set of individual time series data, the objective is to group similar time series into the same cluster" [41].

- Subsequence Clustering: "Given a single time series, sometimes in the form of streaming time series, individual time series (subsequences) are extracted with a sliding window. Clustering is then performed on the extracted time series subsequences" [41].

They then proceed to demonstrate that subsequence clustering is "meaningless" because "clusters extracted from these time series are forced to obey a certain constraints that are pathologically unlikely to be satisfied by any dataset, and because of this, the clusters extracted by any clustering algorithm are essentially random" [41]. In remote sensing, the typical goal is to cluster time-series using all the available data, and this is a direct application of existing constrained clustering approaches. Therefore this work will focus on 'Whole Clustering'.

It should be noted that DTW is not the only alternative to Euclidean distance for measuring dissimilarity between time-series. It is, nevertheless, often found that alternative dissimilarity measures are not significantly better than DTW in real-world datasets [42, 31, 43, 44] (the reader is referred to these references for a comprehensive review and comparison of the alternatives). To quantify the benefit (if any) of using DTW, this study will compare the performance of constrained clustering algorithms modified to use DTW to that using Euclidean.

D. Constrained Clustering

Constrained clustering approaches broadly fall into the following six categories: k-Means, Metric Learning, Spectral Graph Theory, Ensemble, Collaborative, and Declarative.

In k-Means approaches, the clustering algorithm or the objective function is modified so that the user constraints guide the algorithm towards a more appropriate data partitioning. Most of these works consider instance-level ML and CL constraints. The extension is done either by enforcing pairwise constraints, e.g. COP-COBWEB [26], COP-KMeans [13], Seed-KMeans [45], and Constrained-KMeans [45], or by using pairwise constraints to define penalties in the

objective function, e.g. CVQE [29], LCVQE [46], Tabu search [47], and PCK-Means [48]. Each varies in its optimisation criterion and search strategy, for further detail the reader is referred to [49]. A limitation of COP-KMeans is that it does not guarantee convergence and will fail if a constraint is violated.

Metric learning approaches aim to automatically learn a metric measure from training data that best discriminates the data samples. To achieve this in a constrained clustering setting, similar and dissimilar pair sets are constructed using the ML and CL constraints respectively. These are then used as the training data to learn the metric. Examples of this approach using different distance metrics are: Euclidean and shortest path [50], Mahanalobis [51, 52, 53], Kullback-Leibler divergence [54], string-edit [55], and the Laplacian regularizer metric learning (LRML) method for clustering and imagery [56, 57]. As these algorithms are tied to a distance metric, novel algorithms are required for DTW.

Spectral clustering is a non-supervised method that takes as input a pre-calculated similarity matrix and aims to minimise the ratio cut criterion [58] or the normalised cut criterion [59]. Spectral clustering is often considered superior to classical clustering algorithms, such as k-Means, because it is capable of extracting clusters of arbitrary form [58]. It has also been shown that algorithms that build partitions iteratively (like k-Means and EM) are prone to be overly constrained [60]. Moreover, spectral clustering has polynomial time complexity. Algorithms of this type have been proposed by Kamvar et al. [61], Alzate and Suykens [62], Wang and Davidson [63, 64], Wang et al. [65], and Rangapuram and Hein [66].

The abundance of clustering methods found in the literature can be explained by the ill-posed nature of the problem. Indeed, each clustering algorithm is biased by the objective function used to build the clusters. Consequently, different methods can produce very different clustering results from the same data. Furthermore, even the same algorithm can produce different results depending upon its parameters and initialisation. Ensemble clustering methods have the objective of improving the overall quality of the clustering by reducing the bias of each single classification [67]. An ensemble clustering is composed of two steps. First, multiple clusterings are produced from a set of methods having different points of view. These methods can be based on different clustering algorithms [68] or from the same algorithm with different parameter values or initialisations [69]. The final result is derived from the independent results by a consensus function. Constraints can be integrated in two manners: each learning agent integrates them in its own fashion [70, 71, 72]; or applying them in the consensus function [73, 74, 75].

Collaborative clustering is similar to ensemble clustering, but it considers that the information offered by different sources and different clusterings are complementary [76]. Moreover, an important problem encountered by ensemble clustering is the difficulty of computing a consensual result from different results with a wide range of cluster numbers—the correspondence between each cluster is not a trivial problem [77]. Collaborative clustering consists in making multiple clustering methods collaborate to reach an agreement on data partitioning. While ensemble clustering focuses on merging clustering results, collaborative clustering focuses on iteratively modifying the clustering results by sharing information between them [78, 79, 80]. Consequently it extends ensemble clustering by adding a refinement step before the unification of the results. Thus, making the unification easier. In collaborative clustering, three main stages for user constraint integration can be identified [77]: (1) generation of the final result (by labeling the clusters of the final result using label constraints); (2) directly in the collaborative clustering (in order to guide the collaborative process); and (3) using constrained agents.

Declarative approaches offer the user a general framework to formalise the problem by choosing an objective function and explicitly stating the constraints. They enable the modeling of different types of user constraints and the search for an exact solution—a global optimum that satisfies all the user constraints. These frameworks are usually developed using a general optimisation tool, such as integer linear programming (ILP) [81, 82], SAT [83], or constraint programming (CP) [84]. While the other approaches usually focus on instance-level ML and CL constraints, declarative approaches using CP or ILP allow direct integration of cluster-level constraints. They also allow for the integration of different optimisation criteria within the same framework, while other approaches are usually developed for one particular optimisation criterion.

For an in-depth review of the literature on constrained clustering the reader is referred to [14].

III. Constrained Clustering for Satellite Image Time-Series

A subset of the algorithms present in the literature were chosen based upon whether implementations are publicly available (these typically use the Euclidean distance measure) and their ability to be modified to use DTW. The initial implementations were taken from the original authors and all modifications were validated with the authors where possible.

Spectral clustering algorithms take as their input a similarity matrix. This simplifies their modification to use DTW as the similarity matrix can be pre-computed using any distance measure and the methods require no, or little, modification. In the case of spectral methods, the

form of the Laplacian matrix needs consideration. In the presented evaluation, fully connected graphs were used and similarity was calculated using the Gaussian function, such that

$$s_{ij} = \exp\left(\frac{-d_{ij}}{2\sigma^2}\right), \tag{1}$$

where d_{ij} is the DTW dissimilarity between points i and j and σ controls the widths of the neighbourhoods, its value was optimised on each training set using grid search (as were the number of principal components). The algorithms modified in this way were: Adjacency Matrix Modification [61], Constrained 1-Spectral Clustering (C1-SC) [66], Constrained Clustering via Spectral Regularization (CCSR) [85], CSP [65], and Guaranteed Quality Clustering [86].

Declarative approaches can take as their input either the data points or a dissimilarity matrix, depending upon the objective function used. The declarative approach found for this study is CPClustering [84], which takes the pre-computed DTW dissimilarity matrix and therefore does not need modification.

k-Means based algorithms are more involved to apply to time-series as they iteratively calculate distances to cluster centroids and update these centroids. This implies that the algorithm itself needs to be modified to integrate the DTW measure and to use DBA to calculate the cluster centroids. The COP-KMeans [13], LCVQE [46], MPCK-Means [87], Tabu search [47], and MIP-KMeans [82] were found and modified to incorporate these changes.

Collaborative approaches offer several means to integrate constraints. In this study, SAMARAH [77] was modified to use DTW k-Means agents (see above) and pairwise constraints in the collaborative process through its quality criterion, such that

$$Q(\mathcal{C}^{(i)}) = \sum_{c=1}^{N_c} q_c(\mathcal{C}^{(i)})p_c, \tag{2}$$

where N_c is the number of constraints to satisfy, $q_c(\mathcal{C}^{(i)}) \in [0,1]$ is the criterion used to evaluate the result for the c-th constraint and p_c is the weight given by the expert for the c-th constraint ($p_1 + p_2 + \cdots + p_{N_c} = 1$). By default each constraint has the same weight, $1/N_c$.

Of these modified implementations, only a few were found to be suitable to include in the study due to various reasons. The implementation by Wang et al. [65] is formulated for two class problems (although authors describe how the method can be extended to multi-class problems in their article). That by Cucuringu et al. [86] was modified but did not converge on all the datasets. Constrained 1-Spectral Clustering [66], Tabu search [47], and MIP-KMeans [82] were too slow once modified to use DTW and DBA. The method by Pelleg and Baras [46] updates the centroids

heuristically using a formulation which does not have an obvious extension to DTW. Similarly, it is unclear how probabilistic methods such as the constrained EM algorithm [88] could be modified to use alternative distance measures and averaging techniques as it is not obvious how these would affect the probability estimates needed for their application. MPCK-Means [87] implements metric learning, as such is intrinsically tied to norm based distance metrics. A declarative approach using ILP [81] is publicly available and was modified however, it uses a C++ implementation of the COP-KMeans algorithm to pre-compute an initial clustering and currently no C++ implementation of DBA exists in order to modify this implementation.

Implementations of these modified algorithms are available from https://sites.google.com/site/tomalampert/code.

After modification and initial evaluation, the following algorithms were found to be suitable for the remainder of this study: COP-KMeans [13], MIP-KMeans [82], Spec [61] (spectral), CCSR [85] (spectral), CPClustering [84] (declarative), and SAMARAH (using 3 k-Means agents) [77] (collaborative).

A. Complexity

The problem of clustering while fully satisfying constraints is NP-Hard [29]. Indeed the problem of graph colouring, which is NP-Complete, can be reduced in polynomial time to the problem of satisfying cannot-link constraints. Methods such as COP-KMeans, Spec, CCSR, and SAMARAH try to solve the problem heuristically and can therefore violate some constraints (Spec, CCSR, SAMARAH) or get stuck trying to satisfy all the constraints (COP-KMeans). The complexity of COP-KMeans is the complexity of K-means, plus a factor for checking constraint satisfaction. So it is $O(NKtc)$, where N is the number of points, K the number of clusters, t the number of iterations, and c the number of constraints. Spec and CCSR use eigendecomposition so their complexity is $O(KN^2)$. The complexity of SAMARAH is $O(K^2Na(a-1)tc)^*$, where a the number of agents, and c the number of conflicts covered (in these experiments it typically ranged from 200 to 800). Declarative approaches such as CPClustering and MIP-KMeans consider the problem in its entire combinatorial configuration and use general optimisation tools to solve it. Note that CPClustering not only uses the constraint propagation and branch-and-bound mechanisms of CP but it also improves search space pruning by designing dedicated global constraints. Furthermore,

---

*when using the Wemmert-Gançarski cluster quality evaluation metric [89].

(a) Image   (b) Class Labels   (c) Reference Data

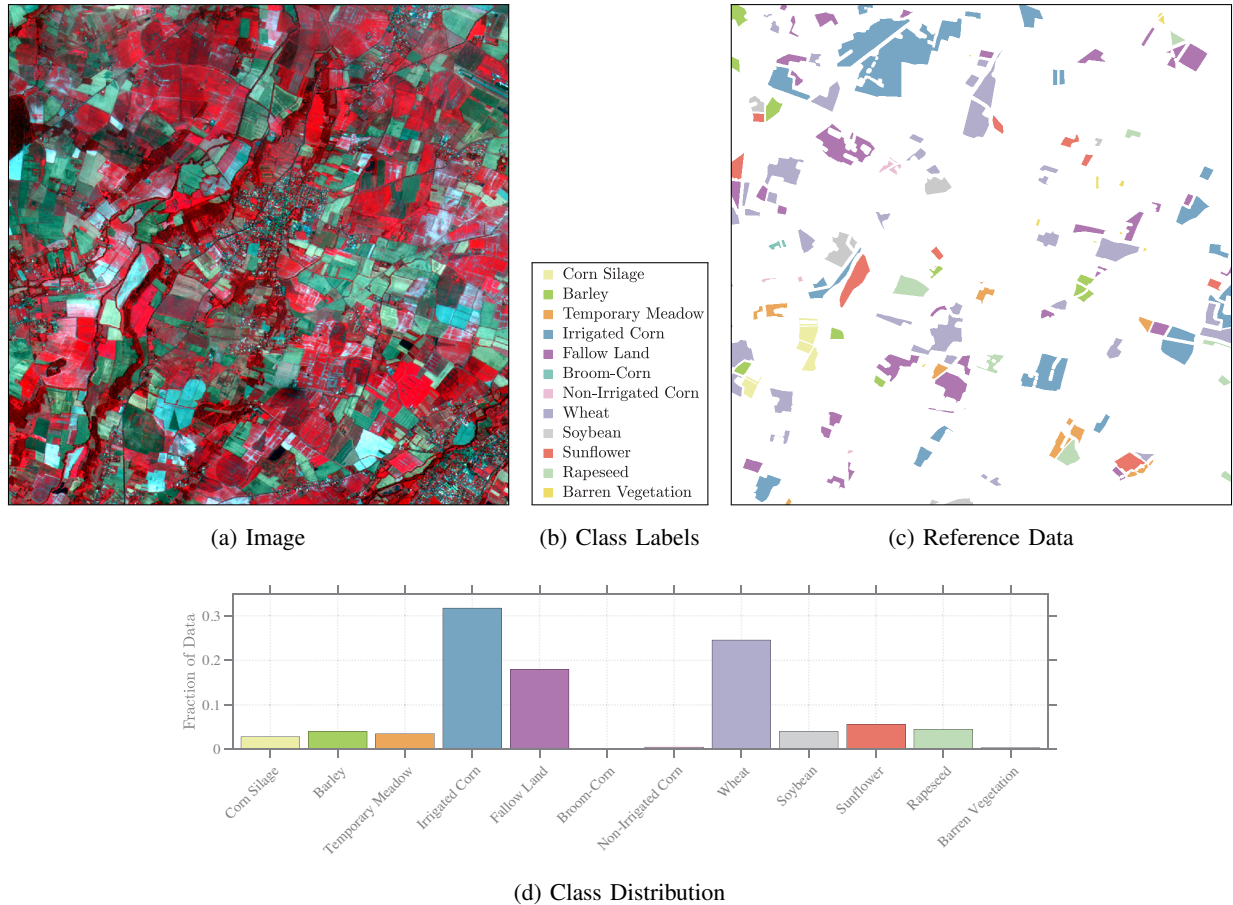|	
■	Corn Silage
■	Barley
■	Temporary Meadow
■	Irrigated Corn
■	Fallow Land
■	Broom-Corn
■	Non-Irrigated Corn
■	Wheat
■	Soybean
■	Sunflower
■	Rapeseed
■	Barren Vegetation

(d) Class Distribution

Figure 2. Real-world image time-series clustering data: 12 classes, and 11 time points (t_4 displayed here).

when using DTW, the methods MIP-KMeans, COP-KMeans, and SAMARAH use DBA therefore its complexity is an additional factor: $O(InT^2)$, where I is the number of iterations, n is the number of sequences to be averaged, and T is their length.

B. Out-of-Sample Classification

Since constrained clustering is a relatively expensive process, it may not be possible to cluster all available data. Using the clustering derived from a subset of the data, out-of-sample data points can be classified using, for example, a k-NN classifier. This allows the generalisation from a randomly selected subset of the data to the whole dataset. If the clustering is to be generalised beyond this, however, the user should be aware of problems arising from distribution mismatches between different domains [90, 91].

IV. EXPERIMENTAL COMPARISON

This section demonstrates the application of constrained clustering to crop classification and tree cut detection. To allow systematic and objective comparison of the algorithms described above, constraints are generated from the ground truth data for crop classification, and the use of user derived constraints is demonstrated in the tree cut problem.

A. Crop Classification

Agricultural practices are a major concern because of their environmental impact, the amount of water they use and the quantity of food they are able to supply [7]. Since satellite data offers observations with a high spectral, spatial and temporal resolution, applications of crop monitoring, identification of agricultural practices, or estimation of yield and crop parameters are numerous [92, 93, 94]. Therefore a dataset that covers cultivated landscape characterised by mixed farming with a wide variety of irrigated and non-irrigated crops is used for evaluating the algorithms. The study area is located near Toulouse (Southwest France). The dataset represents 11 multispectral images non-uniformly sampled over a period of eight months in 2007 (t_1 15th February, t_2 23rd February, t_3 20th April, t_4 30th June, t_5 7th July, t_6 4th August, t_7 11th August, t_8 8th September, t_9 15th September, t_{10} 13th October, and t_{11} 20th October). One of the images in the time-series is presented in Figure 2a. The multipsectral images (green, red, and near-infrared bands) are captured by the Formosat-2 satellite and were provided by *Centre d'Études Spatiales de la Biosphère (CESBIO) Unité Mixte de Recherche CNES-CNRS-IRD-UPS*, Toulouse, France[2]. A random subset of a 1000×1000 region of the image time-series are sampled (within regions for which reference data are available, see Figure 2c) such that the training and test sets contain 1974 and 9869 samples respectively, and where a sample is the evolution of a pixel's spectral values over the 11 images. The training set is used to optimise parameters using grid search for algorithms that require this (Spec and CCSR), and is performed for each experimental setup presented in this section. The experimental results presented are calculated on the test set.

---

[2]It is noted that more recent time-series image data is available, i.e. from the Sentinel-2 satellite; however, at the time of writing, there exists no associated reference data created using temporal evolution. Reference data such as carte d'occupation des sols (COS) exist but are created using supervised methods and therefore make an unfair comparison. This work, therefore, relies on the RPG (registre parcellaire graphique) reference data, for which there are currently no releases that coincide with complete years of Sentinel-2 data. Nevertheless, the methods described are general and their application is not restricted to the type of data used in this evaluation.

Reference data is the farmer's declaration of crops grown to the European Environment Agency for the Common Agricultural Policy.

1) Constraint Generation: To systematically and objectively compare the algorithms, constraints were generated by taking pairs of points randomly and generating a must-link or cannot-link constraint depending upon whether they belonged to the same class or not. Intuitively this is equivalent to a user defining points/regions within the image time-series that contain the same thematic class (corn, for example) and different thematic classes. Different sizes of constraint sets were considered: 5%, 10%, 15%, and 50% of the number of points N in the sampled subsets (they represent a very small fraction of the total number of possible constraints, which is $\frac{1}{2}N(N-1)$). Ten repetitions of each constraint set size were generated, as such each experiment was repeated ten times, each with a different random subset of constraints (all algorithms are evaluated using the same random constraint sets). Finally, the algorithms were executed with no constraints for comparison.

2) Parameter Tuning and Preprocessing: To fully explore the capabilities of the parameterised algorithms, parameter values were optimised by grid search on the training sets (except for CPClustering, which is parameterless). Parameter optimisation is an unrealistic use-case as in real-world applications as a training set is unlikely to be obtainable, however, it demonstrates the theoretical benefits of each algorithm. All the evaluated algorithms, except CPClustering, require the number of clusters K to be set a priori and this was taken from the reference data such that $K = 12$, see Figure 2d (CPClustering relaxes this assumption by accepting a range of K and determining the best value automatically, however in these experiments K was fixed as with the other algorithms).

All samples were normalised to unit length. The adjusted Rand index (ARI) [95] and constraint satisfaction (Sat., the proportion of satisfied constraints) metrics are used to evaluate performance. The presented results represent the mean of ten repetitions of each experiment (each with a different random set of constraints) and are rounded to three decimal places.

To measure the amount of agreement between the objective function and bias of the algorithms and the constraints (referred to as consistency or Con.—the inverse of inconsistency [96] or informativeness [97])—the algorithms were evaluated in an unconstrained setting and the fraction of satisfied constraints determined using the 50% constraint sets.

3) The Value of Time-Series: An algorithm independent measure of cluster (as defined by the reference data) distribution can be evaluated using the mean silhouette score [98]. This score

varies between -1 and 1 and evaluates cluster overlap such that

$$s(o_i) = \frac{b(o_i) - a(o_i)}{\max\{a(o_i), b(o_i)\}}, \tag{3}$$

where $a(o_i)$ is the average dissimilarity of point o_i to all other points within the same cluster and $b(o_i)$ is the minimum, taken over all the clusters C_k not containing o_i, of the average dissimilarity of o_i to all points of C_k. Averaging this over all points in a dataset gives its silhouette score, such that

$$S = \frac{1}{|\mathcal{O}|} \sum_{o_i \in \mathcal{O}} s(o_i). \tag{4}$$

Higher scores indicate that points belong to clusters in which the other points are similar, and are dissimilar to the points in the next closest cluster.

To assess the impact of using a pixel's temporal evolution in agricultural land cover classification, the first analysis is concerned with the pixel values from each individual image (i.e. with no information of their temporal evolution). The Silhouette scores are first assessed. The values for the training sets of individual images are as follows: t_1 -0.191, t_2 -0.148, t_3 -0.054, t_4 -0.314, t_5 -0.272, t_6 -0.003, t_7 -0.010, t_8 0.072, t_9 -0.002, t_{10} -0.247, and t_{11} -0.195. Resulting in a mean Silhouette score of -0.124. The same measure applied to the time-series (and for each distance measure) results in values of: $t_{1,...,11}$ 0.239 (Euclidean) and $t_{1,...,11}$ 0.181 (DTW). It becomes obvious that the information gained by forming a time-series of images results in better separation between the classes and it is expected that the same pattern will be observed in clustering performance.

To verify the above finding, each algorithm is applied without constraints to each of the 11 images in the time series (using the Euclidean distance and training where necessary). The mean ARI achieved in each of the 11 images for each algorithm is presented in Table I. These results demonstrate that Spec results in the best performance in 8 of the 11 images, and when the performance over all images is averaged.

Table II presents the results of unconstrained clustering applied to the time-series ($t_{1,...,11}$) using Euclidean and DTW difference measures in terms of ARI and constraint satisfaction (with respect to the 50% constraint sets) averaged over 10 random sets. This allows the impact of adding a pixel's temporal information into the clustering process to be assessed. It confirms that, in the application of agricultural classification, using a time-series of images outperforms individual image classification by a large margin: ARI is improved by 0.249 when applying the

Table I

UNCONSTRAINED ARI ON THE SUD DATASET USING INDIVIDUAL IMAGES $(t_1, \ldots, t_{11})$—BEST PERFORMANCE FOR EACH IMAGE HIGHLIGHTED IN BOLD.

Method	t_1	t_2	t_3	t_4	t_5	t_6	t_7	t_8	t_9	t_{10}	t_{11}	Mean
COP-KMeans	0.256	0.252	0.283	0.142	0.217	0.361	0.357	0.267	0.225	0.158	0.166	0.244
MIP-KMeans	0.232	0.251	0.296	0.147	0.215	0.407	0.350	0.275	0.217	0.151	0.169	0.246
Spec	**0.567**	**0.588**	**0.659**	0.220	**0.453**	**0.676**	**0.674**	**0.693**	**0.525**	0.172	0.139	**0.488**
CCSR	0.354	0.392	0.511	**0.254**	0.295	0.532	0.510	0.446	0.332	0.142	**0.348**	0.374
CPClustering	0.348	0.412	0.437	0.215	0.400	0.642	0.588	0.492	0.347	**0.199**	0.149	0.385
SAMARAH	0.313	0.286	0.373	0.130	0.198	0.410	0.322	0.222	0.208	0.151	0.177	0.254

Spec algorithm to a time-series compared to the average of all individual images. These results also show that Euclidean distance outperforms DTW in most cases, a result that will be discussed further in the next section.

These results highlight one pitfall of using individual images. Clustering performance is highly dependent upon the time of acquisition. The best performance is observed when the contrast between crops is high (for instance between corn and wheat in August, t_6 and t_7). Lower performance is observed in t_{10} and t_{11} (October) when harvesting has taken place and soil is bare. The best performance is always observed during the period of maximal vegetation (between June and August) and decreases before and after these periods. These trends can also be observed in the corresponding Silhouette scores.

4) The Benefit of Constraints: The unconstrained time-series ARIs (Table II) will serve as a baseline to gauge any improvements when introducing constraints. The mean ARIs and Constraint Satisfaction results for the constrained clustering experiments are presented in Table III. Again, Spec results in the highest mean ARI when using the DTW distance measure, however, SAMARAH sees the largest increase in performance when constraints are integrated and surpasses Spec in two constraint fractions when using the Euclidean distance measure. CPClustering offers the greatest constraint satisfaction and clustering performance.

Generally, the performance of clustering time-series is improved by integrating constraints, however, it is not universally observed to be true. The most consistent improvements in ARI (irrespective of distance measure) are observed with methods that do not guarantee constraint satisfaction (Spec, CCSR, and SAMARAH). Those that do (COP-KMeans, MIP-KMeans, and

CPClustering) improve their rate of constraint satisfaction when compared to the unconstrained case. More often than not this leads to a reduction in ARI as the algorithms struggle to balance what is assumed to be conflicting information from the constraints and the distance measure, also explaining why ARI improvements are observed with algorithms that can violate constraints. The conflict between constraints and the distance measure will be further explored in the next subsection.

Table II

UNCONSTRAINED ARI & CONSISTENCY ON THE SUD DATASET. EACH MEASURE'S BEST PERFORMANCE IS BOLD AND OVERALL UNDERLINED.

Method	Distance	ARI	Sat.
COP-KMeans	DTW	0.426	0.812
	Euclidean	0.420	0.807
MIP-KMeans	DTW	0.398	0.805
	Euclidean	0.407	0.803
Spec	DTW	**0.531**	**0.840**
	Euclidean	<u>**0.737**</u>	<u>**0.885**</u>
CCSR	DTW	0.359	0.797
	Euclidean	0.450	0.817
CPClustering	DTW	0.437	0.803
	Euclidean	0.681	0.413
SAMARAH	DTW	0.406	0.802
	Euclidean	0.463	0.817

5) Out-of-Sample Classification: The methodology described in Section III-B allows us to classify all the labelled points in the 1000×1000 pixel time-series (i.e those presented in Figure 2c) using the output of the constrained clustering process applied to the test sets. To this end, the output of SAMARAH (which achieves the highest performance without requiring parameter tuning) with the median ARI for both distances were determined. These represent a realistic expectation of applying the algorithm with an arbitrary constraint set, and as such formed training sets for a k-NN classifier, where $k = 3$. The results obtained (represented as an image) are presented in Figure 3, along with the contingency tables associated with the clusterings used as training sets. This process does result in a loss of accuracy when compared to the original

Table III

CONSTRAINED ARI AND CONSTRAINT SATISFACTION ON THE SUD DATASET. THE BEST PERFORMANCES FOR EACH MEASURE, CONSTRAINT FRACTION, AND DISTANCE MEASURE ARE HIGHLIGHTED IN BOLD.

Method	Distance	5%		10%		15%		50%	
		ARI	Sat.	ARI	Sat.	ARI	Sat.	ARI	Sat.
COP-KMeans	DTW	0.416	**1.000**	0.398	**1.000**	0.407	**1.000**	0.436	**1.000**
	Euclidean	0.406	**1.000**	0.414	**1.000**	0.443	**1.000**	0.369	**1.000**
MIP-KMeans	DTW	0.381	**1.000**	0.390	**1.000**	0.366	**1.000**	0.433	**1.000**
	Euclidean	0.428	**1.000**	0.416	**1.000**	0.431	**1.000**	0.532	**1.000**
Spec	DTW	**0.683**	0.867	**0.784**	0.922	**0.725**	0.888	**0.786**	0.911
	Euclidean	0.671	0.854	**0.730**	0.900	0.702	0.875	**0.781**	0.916
CCSR	DTW	0.578	0.858	0.536	0.843	0.528	0.838	0.527	0.840
	Euclidean	0.658	0.878	0.535	0.843	0.535	0.842	0.503	0.835
CPClustering	DTW	0.469	**1.000**	0.513	**1.000**	0.510	**1.000**	0.589	**1.000**
	Euclidean	0.650	**1.000**	0.562	**1.000**	0.542	**1.000**	0.510	**1.000**
SAMARAH	DTW	0.597	0.870	0.657	0.876	0.637	0.867	0.681	0.878
	Euclidean	**0.691**	0.884	0.682	0.884	**0.714**	0.890	0.702	0.885

clustering: ARI decreases by 0.172 (0.154) and 0.191 (0.173) when using the DTW and Euclidean distances respectively (figures in brackets include the original clustering results). This is to be expected when generalising to unseen data. It is clear that the clustering-classifications result in homogeneous regions without having any information regarding spatial proximity nor constraints preserving spatial smoothness. The contingency tables of the clusterings used as training data, Figures 3c (DTW) and 3d (Euclidean), indicate that, while the classes are generally distinctly separated into specific clusters, several classes may be represented by single clusters. For example in Figure 3c cluster 2 contains high proportions of the Irrigated Corn, Soybean, and Rapeseed classes. This tendency appears to be reduced when using the Euclidean distance (Figure 3d) and it is hypothesised that this is because DTW attempts to find an optimal warping between two time-series, which may distort the distances between thematic classes.

Figure IV-A5 focuses on one region of the image classification using the highest ARI clusterings for each algorithm. It reiterates that SAMARAH results in regions that are homogeneously classified as does the Spec algorithm, which also achieves high ARIs. This is confirmed by measuring

(over the whole image, not just the crop displayed in Figure IV-A5) the ratio of the number of pixels of the largest class in each region (defined by the reference data) to that of all other classes in the same region (therefore a value of $1/N$, where N is the number of classes, means that all the classes are evenly distributed in a region, and 1 means that a region is made up of only one class, the value for each region is weighted relative to its size): COP-KMeans 0.602, MIP-KMeans 0.694, Spec 0.913, CCSR 0.663, SAMARAH 0.802, and CPClustering 0.525.

A more detailed view of the clustering can be seen in the contingency tables presented in Figures 3c and 3d. These indicate that the clusterings are similar whether taking the maximum or median ARI, indeed the difference between the maximum and median ARIs is 0.074 when using Euclidean distance and 0.051 when using DTW.

It should be noted that the reference data used in the problem was not perfect, as is common in large scale remote sensing problems. Figures 3e–3l highlight this by focusing on one region of the reference data. It can be observed that although the reference data indicates one homogeneous region, the data does not—what appear to be paths and tracks cross the field. This complicates the clustering algorithm's objective function and causes a conflict between the constraints and the information derived from the distance measure. This is particularly disadvantageous for algorithms that satisfy all constraints (CPClustering, COP-KMeans, and MIP-KMeans), and could have contributed to their lower ARI scores (a pre-processing stage to remove conflicting constraints may help to avoid this). This conflict contributes to the confusion between clusters and classes observed in the contingency tables.

B. Tree Cut Detection

Thus far the process of obtaining constraints has been simulated using ground truth data. This section demonstrates the use of user derived constraints and another real-world application that uses the more recent Sentinel-2 data.

There is currently a high demand for ecologically derived timber and there is a large amount of tree felling in European forests. Even though a large number of trees remain, replanting operations are essential for sustainability and the time scale associated with tree growth requires early detection of felling activities. A body of the literature dedicated to detecting tree 'clear-cuts' [23, 24] exists and there is recent political interest in monitoring such activities, e.g. the

RegioWood II initiative[3].

An area of the Vosges Mountains (Alsace, France), 724×337 pixels in size, was selected for this study, see Figure 4a. The time-series is composed of 10 images sampled over 3 years: t_1 23rd August 2016, t_2 26th August 2016, t_3 10th May 2017, t_4 19th June 2017, t_5 23rd August 2017, t_6 25th September 2017, t_7 8th May 2018, t_8 2nd July 2018, t_9 24th July 2018 and $t_1 0$ 16th August 2018. The dataset contains 10 areas of clear-cut: 8 appear at t_3 and 2 at t_7. The images are composed of one band that contains the computed NDVI (normalised difference vegetation index) values. The images were computed from Copernicus Sentinel-2 data (tile T32ULU) processed at level 2A/3A by CNES for Theia Land Data Centre.

A subset of the area was selected by randomly sampling $30,000$ pixel time-series. The ground truth locations of clear-cuts used for validation was manually created by remote sensing experts and validated in the field by forestry rangers, see Figure 4b, one of the challenges here being that clear-cuts represent only 0.3% of the dataset. Furthermore, as is typical of remote sensing problems, the number of clusters that represent the non-clear-cuts part of the image is unknown. As such, the number of clusters is arbitrarily set to $k = 15$. The evaluation is only concerned with clusters that include clear-cuts (this is selected by choosing the cluster the overlaps the clear-cut ground truth the most). As before, SAMARAH was used with both the DTW and Euclidean distance measures.

The constraints were given by an expert in remote sensing after viewing an out-of-sample classification of the whole image based on the unconstrained clustering of the subset described above, see Figure 4c. One set was given using the DTW metric, and another using the Euclidean metric (the number of constraints remained constant). If the added constraints concerned a sample that was not in the original dataset, the sample was added in order to guide SAMARAH to extract meaningful information (which resulted in $30,037$ samples for DTW and $30,034$ for Euclidean). The average recall and precision (of 10 repetitions, standard deviations are presented in brackets) of the unconstrained clusterings for DTW was 0.367 (0.138) and 0.018 (0.009) respectively and for Euclidean was 0.558 (0.141) and 0.040 (0.012) respectively.

The low precision is evident in Figure 4c, in which it can be seen that the unconstrained DTW clustering confuses harvested meadows with tree cuts because both exhibit a sharp decrease in NDVI (the Euclidean version of these figures are omitted due to space restrictions). The expert

---

[3]http://www.regiowood2.info

is able to use his background knowledge to identify this confusion and add CL constraints to reduce it, see Figure 4d. Examples of the samples that are bound by these CL constraints are presented in Figure 4f, which show the similarities between the tree-cut samples (dashed lines) and those of harvested meadows (solid lines).

When 19 constraints are added, the average recall and precision (of 10 repetitions) increase to 0.634 (0.123) and 0.294 (0.145) respectively for DTW and 0.578 (0.0769) and 0.421 (0.304) respectively for Euclidean. DTW achieves a better recall at the expense of precision, and Euclidean the inverse. Notably, the standard deviation of precision increases when using Euclidean. This is because 7 of the 10 repetitions had a cluster containing the cuts occurring at t_3, which are the majority of the cuts, giving good recall (0.534 on average) and precision (0.567 on average); but the remaining 3 were not meaningful for the task at hand (a precision of 0.081 on average). Another experiment was conducted using the Euclidean version to try to merge the cuts occurring in both dates into one cluster by modifying 3 ML constraints (out of 9). This was successful in most (9 out of 10) of the clusterings and increased recall to 0.698 (0.0122), but also caused a large amount of noise to be introduced into the cluster, resulting in a precision of 0.136 (0.216). This is because the algorithm is forced to try to resolve the conflicting information derived from the distance measure and data, with that given by the user, and highlights the necessity to select the correct distance measure for the problem to be tackled.

Assuming that this is the case, it can be observed that relatively few constraints lead to a large increase in performance. This is in contrast to the previous experiments, and is a result of the expert having the opportunity to choose constraints that are meaningful to the algorithm being used. For example, these constraints are more effective than adding CL constraints between tree cuts and road or water as these are easily separated by the algorithm's objective function. This type of constraint is referred to as being 'consistent' [14], although there is no way to predict consistency a priori.

The false positives observed in Figure 4f can be grouped into three sources: the northern limit of forested areas, because shadows and canopy variation between seasons can be misinterpreted in a pixel based approach; the exact border of regions of tree cuts is not well defined and so the algorithm may detect variances that are not covered by the GT; and some harvested meadow pixels remain in the tree cut cluster because their NDVI signals are similar.

V. Practical Considerations

This section offers further analysis of the results and discusses their implications. Since the objective function of each method is different, comparisons are based on several different aspects: the facility of use, the adaptability of the method, quality of the returned solution, and run time.

A. Parameter Values

The choice of parameter values for each algorithm is an important factor to their performance. By using CPClustering this choice is avoided as it is parameterless. Algorithms based on k-Means (COP-KMeans, MIP-KMeans, and SAMARAH) are relatively simple to apply as it is only necessary to choose a sufficiently large number of iterations. Spectral methods, on the other hand, require at least two parameter values to be chosen: σ, which controls the connectivity in the similarity graph, and the number of eigenvectors used for clustering. These parameters are not intuitive to the user and the algorithm's performance is highly dependent upon their choice [58].

Figure 5 presents the variation of training ARI in the Crop Classification problem when different parameter combinations are chosen during the grid search for the Spec and CCSR algorithms, and explains the high ARIs the spectral methods achieved in the previous section. These were largely due to good parameter values. Without the prior knowledge that is offered by a training set (i.e. in an semi-supervised setting) a user is unlikely to choose these 'optimal' values, and straying from these results in a large reduction in clustering performance. The mean ARI obtained for each algorithm are 0.367 and 0.297 (Spec and CCSR respectively), which are considerably lower than the maximum values of 0.720 and 0.466 (Spec and CCSR respectively). It should be noted that works exist to remove these parameter choices for unconstrained spectral clustering [99, 100].

B. Distance Measure

It was found in the Crop Classification dataset that the Euclidean distance offers the highest unconstrained performance in almost all of the algorithms. This is due to the relatively coherent nature of the data. The images are cloud free, which means that the feature vectors are noise free. The problem also lends itself to the Euclidean distance: the time-series are aligned in the time domain and the area of interest is spatially coherent, meaning that crops grow and are harvested

in the same time-frame within different parts of the image. If these assumptions do not hold, DTW may offer better performance.

From the Tree Cut dataset it can be concluded that if the phenomenon to be clustered has a temporal signature that varies between samples (e.g. cuts occurring at different dates) and these should be combined, DTW would be a better choice, otherwise (if the signals are temporally correlated and/or the uncorrelated signals should be separated) Euclidean can give better results.

To the best of our knowledge, however, the theoretical guarantees of the K-Means algorithm only hold when using the Euclidean distance. DTW barycenter averaging is a stochastic process and therefore the same guarantees are unlikely to hold when using it, nor does DTW satisfy the triangle inequality.

C. Run-Time

Table IV

EXECUTION TIME (IN SECONDS) OF THE CROP CLASSIFICATION DATASET MEASURED USING AN INTEL$^{®}$ CORE™ I5-4670 CPU @ 3.40GHz PROCESSOR WITH 32GB RAM. * INDICATES THAT THE METHOD'S TIMES EXCLUDE THE DISTANCE MATRIX COMPUTATION. THE NUMBER OF CPCLUSTERING'S GLOBALLY OPTIMAL SOLUTIONS ARE IN BRACKETS (FOUND WITHIN 30 MINUTES).

Method	Distance	0%	5%	50%
COP-KMeans	DTW	1203.093	1185.972	1318.606
	Euclidean	23.552	19.278	19.698
MIP-KMeans	DTW	667.945	660.196	819.458
	Euclidean	56.343	121.086	226.788
Spec*	DTW	214.443	217.662	215.985
	Euclidean	206.655	205.387	213.805
CCSR*	DTW	234.380	233.741	235.311
	Euclidean	219.648	223.818	231.761
CPClustering*	DTW	1745.491 (0)	98.333 (0)	35.767 (1)
	Euclidean	732.185 (0)	294.982 (0)	62.722 (1)
SAMARAH	DTW	1547.156	1237.635	1274.350
	Euclidean	229.509	130.655	114.876

An important aspect of the usability of these methods is the time taken to obtain a clustering. Table IV presents the time taken to execute the clustering algorithm for each implementation

in the Crop Classification problem, averaged over 10 repetitions (each with different random constraint sets) for the unconstrained, 5% and 50% constraint sets. It should be noted that not all algorithms are implemented in the same language, and the timings are taken using each language's built-in timing. Finally, these timings only include the execution of the clustering algorithm itself and, as such, exclude processes such as reading data files. Furthermore, it excludes the training phase needed to select parameter values for Spec and CCSR and the time taken to calculate the distance matrix that is the input for Spec, CCSR, and CPClustering. CPClustering guarantees that the solution does not violate any constraints, it also attempts to find the globally optimal solution by branch and bound, as such a timeout is given (in these experiments 30 minutes) at which point the current best solution is returned. We therefore report the mean time taken to find the returned solution, and the number of globally optimal solutions returned.

These timings demonstrate that the speed of spectral methods (Spec and CCSR) are constant relative to the number of constraints (this is expected as the same computation is performed whether constraints are included or not). The K-Means based algorithms (COP-KMeans, MIP-KMeans, and SAMARAH) are much quicker when using the Euclidean distance, however, because of the repeated distance calculations and averaging, become very slow when using the DTW dissimilarity measure. There is also a slight trend for the timings of MIP-KMeans to increase as more constraints are added. SAMARAH's run time is, however, reduced when constraints are added. This is because any constraints that become unsatisfied cause the solution's quality to degrade, which may halt the algorithm and cause the current best solution to be returned. CPClustering finds an initial solution very quickly, however, it did not find the global optimum in most of the experiments within the timeout. To put this into perspective, CPClustering is the only method that searches for a global optimum and therefore this should not be regarded as a drawback. The large timings in the unconstrained experiments, which reduce when constraints are added, are due to the added complexity of finding solutions that do not violate the constraints. In CPClustering's case the figures represent the time taken to find the solution that is returned at the end of the time out. When constraints are added it is more difficult to update the (rapidly found) initial solution, meaning that it is returned. When the problem is unconstrained the solution is frequently updated, resulting in longer run times. With such quick computation it still offers better quality solutions than the K-Means based algorithms and comparable quality to CCSR (with parameter optimisation) in most cases.

D. Additional Challenges

This study has focused on using constraints in satellite image time-series clustering, however there exist numerous challenges related to capturing these constraints. For example, it is necessary to study and detail the thematic constraints (i.e. the opinions of the expert) to be captured to guide the process.

These thematic constraints can be extremely broad and have to be translated into actionable constraints. In the current state of knowledge, a limited number of actionable constraints into which thematic constraints can be translated exist: ML, CL, label, number of clusters k, and size. The following observations are therefore translatable: "these two objects seem to be of the same nature"; "these two ensembles of objects are of the same nature" (ML constraints between all the pairs of objects of the two sets); "these three sets are completely different" (CL constraint on all the pairs of objects from the three sets); "this object is of type X" (labeling constraint); and "a cluster cannot represent more than 20% of the data" (cluster size constraint). Nevertheless, only a small subset of constrained clustering algorithms can handle all such actionable constraints (e.g. declarative and SAMARAH).

Furthermore, generating actionable constraints from a set of data points can rapidly lead to a significant increase in both combinatorial complexity and the scope of the constraints. For example, a constraint "of the same nature" on two sets of size N_1 and N_2 respectively will, using the naive approach, generate $N_1(N_1 - 1) + N_2(N_2 - 1) + N_1 \times N_2$ ML constraints. These are transitive [13] however, and therefore the number of constraints needed to satisfy the user requirement can be reduced to $(N_1 + N_2) - 1$ under the assumption of guaranteed constraint satisfaction. Nevertheless, this can still result in a very large number of constraints, which could prevent the use of declarative methods. Another example is a constraint that states two sets of size N_1 and N_2 are "of different nature". Depending on the context this constraint can correspond to a disjunction of several sets of constraints.

The following two problems are therefore identified. For each problem we give some direction on how it can be tackled.

- How should algorithms be modified for large constraint sets?
  - Reduce the model's size through the number of considered objects, e.g. sampling or removing irrelevant objects.
  - Relax the solution's optimality via an execution time threshold (with no guarantees of the

result's quality).

  – Searching locally, instead of globally (already possible with CPClustering but not obvious for the others).

- How can the number of constraints be reduced/limited without loss of quality (i.e. define a minimal set of constraints)?

  – Sample the constraints and/or the objects under constraint.

  – Identification of informative constraints.

VI. CONCLUSIONS

To recapitulate, this manuscript has presented a background of constrained clustering and data analysis for satellite image time-series clustering. Limitations of publicly available implementations have been discussed, and those which are able to be modified to use the DTW dissimilarity measure and DBA averaging method were evaluated. A comparative study of these approaches applied to agricultural classification has been conducted and the results analysed. This investigation has been concluded by a discussion of the issues raised.

It has been shown that the effectiveness of the Euclidean and DTW distance measures depend on the problem being tackled and the user's requirements. K-Means based algorithms do not achieve state-of-the-art performance and require modification to use DTW dissimilarity. Collaborative algorithms (e.g. SAMARAH) improve upon this by guiding several unsupervised algorithms. Spectral clustering algorithms, although achieving the highest performance, are limited by the requirement of carefully choosing parameter values, which is unrealistic in semi-supervised settings. This problem is mitigated by declarative approaches, which are parameterless and provide solutions that satisfy all the constraints. They are, therefore, more sensitive to conflicts between the data and constraints.

Challenges in obtaining and using constraints in remote-sensing have been discussed, particularly the explosive nature of constraints being derived from expert opinions and several areas for algorithm development have been outlined.

REFERENCES

[1] N. Li, A. Martin, and R. Estival, "Combination of supervised learning and unsupervised learning based on object association for land cover classification," in *DICTA*, 2018.

[2] G. Cao, X. Li, and L. Zhou, "Unsupervised change detection in high spatial resolution remote sensing images based on a conditional random field model," *Eur J Remote Sens*, vol. 49, pp. 225–237, 2016.

[3] R. Flamary, M. Fauvel, M. D. Mura, and S. Valero, "Analysis of multi-temporal classification techniques for forecasting image times series," *IEEE Geosci Remote S*, vol. 12, no. 5, pp. 953–957, 2018.

[4] L. Khiali, M. Ndiath, S. Alleaume, D. Ienco, K. Ose, and M. Teisseire, "Detection of spatio-temporal evolutions on multi-annual satellite image time series: A clustering based approach," *Int J Appl Earth Obs Geoinf*, vol. 74, pp. 103–119, 2019.

[5] M. Rußwurm and M. Körner, "Temporal vegetation modelling using long short-term memory networks for crop identification from medium-resolution multi-spectral satellite images," in *CVPR*, 2017.

[6] P. Fu and Q. Weng, "A time series analysis of urbanization induced land use and land cover change and its impact on land surface temperature with landsat imagery," *Eur J Remote Sens*, vol. 175, pp. 205–214, 2016.

[7] C. Marais Sicre, J. Inglada, R. Fieuzal, F. Baup, S. Valero, J. Cros, M. Huc, and V. Demarez, "Early detection of summer crops using high spatial resolution optical image time series," *Remote Sensing*, vol. 8, no. 7, p. 591, 2016.

[8] H. Yin, D. Pflugmacher, R. Kennedy, D. Sulla-Menashe, and P. Hostert, "Mapping annual land use and land cover changes using MODIS time series," *European Journal of Remote Sensing*, vol. 7, no. 8, pp. 3421–3427, 2014.

[9] X. Yang and C. Lo, "Using a time series of satellite imagery to detect land use and land cover changes in the Atlanta, Georgia metropolitan area," *Int J Remote Sens*, vol. 23, no. 9, pp. 1775–1798, 2002.

[10] D. Rey, M. Walvoord, B. Minsley, J. Rover, and K. Singha, "Investigating lake-area dynamics across a permafrost-thaw spectrum using airborne electromagnetic surveys and remote sensing time-series data in Yukon Flats, Alaska," *Environ Res Lett*, vol. 14, p. 025001, 2019.

[11] J. Cihlar, R. Latifovic, and J. Beaubien, "A comparison of clustering strategies for unsupervised classification," *Can J Remote Sens*, vol. 26, pp. 446–454, 2000.

[12] J. Senthilnath, S. Omkar, V. Mani, N. Tejovanth, P. Diwakar, and A. Shenoy, "Hierarchical clustering algorithm for land cover mapping using satellite images," *JSTARS*, vol. 5, pp. 762–768, 2012.

[13] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl, "Constrained k-means clustering with background knowledge," in *ICML*, 2001, pp. 577–584.

[14] T. Lampert, B. Lafabregue, T.-B.-H. Dao, N. Serrette, G. Forestier, B. Crémilleux, C. Vrain, and P. Gançarski, "Constrained distance based clustering for time-series: A comparative and experimental study," *Data Min Knowl Discov*, vol. 32, no. 6, pp. 1663–1707, 2018.

[15] S. Siachalou, G. Mallinis, and M. Tsakiri-Strati, "A hidden markov models approach for crop classification: Linking crop phenology to time series of multi-sensor remote sensing data," *Remote Sens*, vol. 7, no. 4, pp. 3633–3650, 2015.

[16] B. Wardlow, S. Egbert, and J. Kastens, "Analysis of time-series modis 250 m vegetation index data for crop classification in the u.s. central great plains," *Remote Sensing of Environment*, vol. 108, pp. 290–310, 2007.

[17] D. Arvor, M. Jonathan, M. Meirelles, V. Dubreuil, and L. Durieux, "Classification of modis evi time series for crop mapping in the state of mato grosso, brazil," *Int J Remote Sens*, vol. 32, pp. 7847–7871, 2011.

[18] H. Carrão, P. Gonçalves, and M. Caetano, "Contribution of multispectral and multitemporal information from modis images to land cover classification," *Remote Sensing of Environment*, vol. 112, pp. 986–997, 2008.

[19] S. Foerster, K. Kaden, M. Foerster, and S. Itzerott, "Crop type mapping using spectral-temporal profiles and phenological information," *Computers and Electronics in Agriculture*, vol. 89, pp. 30–40, 2012.

[20] K. Jia, S. Liang, X. Wei, Y. Yao, Y. Su, B. Jiang, and X. Wang, "Land cover classification of landsat data with phenological features extracted from time series MODIS NDVI data," *Remote Sens*, vol. 6, pp. 11 518–11 532, 2014.

[21] T. Murakami, S. Ogawa, N. Ishitsuka, K. Kumagai, and G. Saito, "Crop discrimination with multitemporal SPOT/HRV data in the Saga plains, Japan," *Int J Remote Sens*, vol. 22, pp. 1335–1348, 2001.

[22] L. Zhong, P. Gong, and G. Biging, "Efficient corn and soybean mapping with temporal

extendability: A multi-year experiment using landsat imagery," *Remote Sensing of Environment*, vol. 140, pp. 1–13, 2014.

[23] T. Saksa, J. Uuttera, T. Kolström, M. Lehikoinen, A. Pekkarinen, and V. Sarvi, "Clear-cut detection in boreal forest aided by remote sensing," *Scandinavian Journal of Forest Research*, vol. 18, pp. 537–546, 2003.

[24] W. Cohen, M. Fiorella, J. Gray, E. Helmer, and K. Anderson, "An efficient and accurate method for mapping forest clearcuts in the pacific northwest using landsat imagery," *Photogramm Eng Remote Sensing*, vol. 64, pp. 293–299, 1998.

[25] P. Hansen and B. Jaumard, "Cluster analysis and mathematical programming," *Mathematical Programming*, vol. 79, no. 1–3, pp. 191–215, 1997.

[26] K. Wagstaff and C. Cardie, "Clustering with instance-level constraints," in *ICML*, 2000, pp. 1103–1110.

[27] S. Anand, D. Bell, and J. Hughes, "The role of domain knowledge in data mining," in *CIKM*, 1995, pp. 37–43.

[28] S. Basu, I. Davidson, and K. Wagstaff, *Constrained Clustering: Advances in Algorithms, Theory, and Applications*, 1st ed.   Chapman & Hall/CRC, 2008.

[29] I. Davidson and S. Ravi, "Clustering with constraints: Feasibility issues and the k-means algorithm," in *SDM*, 2005, pp. 138–149.

[30] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: a survey and empirical demonstration," *DAMI*, vol. 7, pp. 349–371, 2003.

[31] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *DAMI*, vol. 26, no. 2, pp. 275–309, 2013.

[32] H. Sakoe and S. Chiba, "A dynamic programming approach to continuous speech recognition," in *ICA*, vol. 3, 1971, pp. 65–69.

[33] ——, "Dynamic programming algorithm optimization for spoken word recognition," *IEEE Trans Acoust*, vol. 26, no. 1, pp. 43–49, 1978.

[34] F. Petitjean, A. Ketterlin, and P. Gançarski, "A global averaging method for dynamic time warping, with applications to clustering," *Pattern Recognit*, vol. 44, no. 3, pp. 678–693, 2011.

[35] S. Laxman and P. Sastry, "A survey of temporal data mining," *Sadhana*, vol. 31, no. 2, pp. 173–198, 2006.

[36] V. Kavitha and M. Punithavalli, "Clustering time series data stream—a literature survey," *IJCSIS*, vol. 8, no. 1, pp. 289–294, 2010.

[37] C. Antunes and A. Oliveira, "Temporal data mining: an overview," in *RDM*, 2001, pp. 1–13.

[38] T. Warren Liao, "Clustering of time series data—a survey," *Pattern Recognit*, vol. 38, no. 11, pp. 1857–1874, 2005.

[39] S. Rani and G. Sikka, "Recent techniques of clustering of time series data: a survey," *International Journal of Computer Applications*, vol. 52, no. 15, pp. 1–9, 2012.

[40] S. Aghabozorgi, A. Shirkhorshidi, and T. Wah, "Time-series clustering—a decade review," *Information Systems*, vol. 53, pp. 16–38, 2015.

[41] E. Keogh and J. Lin, "Clustering of time-series subsequences is meaningless: implications for previous and future research," *Knowl Inf Syst*, vol. 8, pp. 154–177, 2005.

[42] H. Ding, G. Trajcevski, P. Scheuermann, X. Wang, and E. Keogh, "Querying and mining of time series data: experimental comparison of representations and distance measures," in *VLDB*, 2008.

[43] J. Lines and A. Bagnall, "Time series classification with ensembles of elastic distance measures," *Data Min Knowl Discov*, vol. 29, no. 3, pp. 565–592, 2015.

[44] A. Bagnall, J. Lines1, A. Bostrom, J. Large, and E. Keogh, "The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances," *Data Min Knowl Discov*, vol. 31, pp. 606–660, 2017.

[45] S. Basu, A. Banerjee, and R. Mooney, "Semi-supervised clustering by seeding," in *ICML*, 2002, pp. 19–26.

[46] D. Pelleg and D. Baras, "K-means with large and noisy constraint sets," in *ECML*, 2007, pp. 674–682.

[47] T. Hiep, N. Duc, and B. Trung, "Local search approach for the pairwise constrained clustering problem," in *SoICT*, 2016, pp. 115–122.

[48] S. Basu, A. Banerjee, and R. Mooney, "Active semi-supervision for pairwise constrained clustering," in *SDM*, 2004, pp. 333–344.

[49] I. Davidson and S. Basu, "A survey of clustering with instance level constraints," *ACM Trans Knowl Discov Data*, vol. 77, no. 1, pp. 1–41, 2007.

[50] D. Klein, S. Kamvar, and C. Manning, "From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering," in *ICML*, 2002, pp.

307–314.

[51] A. Bar-Hillel, T. Hertz, M. Shental, and D. Weinshall, "Learning a Mahalanobis metric from equivalence constraints," *JMLR*, vol. 6, pp. 937–965, 2005.

[52] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall, "Learning distance functions using equivalence relations," in *ICML*, 2003, pp. 11–18.

[53] E. Xing, A. Ng, M. Jordan, and S. Russell, "Distance metric learning learning, with application to clustering with side-information," in *NIPS*, 2002, pp. 521–528.

[54] D. Cohn, R. Caruana, and A. Mccallum, "Semi-supervised clustering with user feedback," Dep of Comp Sci, Cornell University, Tech. Rep. TR2003-1892, 2003.

[55] M. Bilenko and R. Mooney, "Adaptive duplicate detection using learnable string similarity measures," in *SIGKDD*, 2003, pp. 39–48.

[56] S. Hoi, W. Liu, and S.-F. Chang, "Semi-supervised distance metric learning for collaborative image retrieval," in *CVPR*, 2008.

[57] ——, "Semi-supervised distance metric learning for collaborative image retrieval and clustering," *ACM Trans Multimed Comput Commun Appl*, vol. 6, no. 3, 2010.

[58] U. von Luxburg, "A tutorial on spectral clustering," *Stat Comput*, vol. 17, no. 4, pp. 395–416, 2007.

[59] J. Shi and J. Malik, "Normalized cuts and image segmentation," *PAMI*, vol. 22, no. 8, pp. 888–905, 2000.

[60] I. Davidson and S. Ravi, "Identifying and generating easy sets of constraints for clustering," in *AAAI*, 2006, pp. 336–341.

[61] S. Kamvar, D. Klein, and C. Manning, "Spectral learning," in *IJCAI*, 2003, pp. 561–566.

[62] C. Alzate and J. Suykens, "A regularized formulation for spectral clustering with pairwise constraints," in *NIPS*, 2009, pp. 141–148.

[63] X. Wang and I. Davidson, "Active spectral clustering," in *ICDM*, 2010, pp. 561–568.

[64] ——, "Flexible constrained spectral clustering," in *KDD*, 2010, pp. 563–572.

[65] X. Wang, B. Qian, and I. Davidson, "On constrained spectral clustering and its applications," *Data Min Knowl Discov*, vol. 28, no. 1, pp. 1–30, 2014.

[66] S. Rangapuram and M. Hein, "Constrained 1-spectral clustering," in *AIStats*, 2012, pp. 1143–1151.

[67] S. Hadjitodorov and L. Kuncheva, "Selecting diversifying heuristics for cluster ensembles," *MCS*, pp. 200—209, 2007.

[68] A. Strehl and J. Ghosh, "Cluster ensembles—a knowledge reuse framework for combining multiple partitions," *JMLR*, vol. 3, pp. 583–617, 2002.

[69] A. Fred and A. Jain, "Data clustering using evidence accumulation," *ICPR*, pp. 276—280, 2002.

[70] Z. Yu, H. Wongb, J. You, Q. Yang, and H. Liao, "Knowledge based cluster ensemble for cancer discovery from biomolecular data," *IEEE Trans Nanobioscience*, vol. 10, no. 2, pp. 76—85, 2011.

[71] Y. Yang, W. Tan, T. Li, and D. Ruan, "Consensus clustering based on constrained self-organizing map and improved cop-kmeans ensemble in intelligent decision support systems," *Knowl-Based Syst*, vol. 32, pp. 101—115, 2012.

[72] A. Iqbal, A. Moh'd, and Z. Zhan, "Semi-supervised clustering ensemble by voting," in *ICIC*, 2012, pp. 1–5.

[73] M. Al-Razgan and C. Domeniconi, *Clustering Ensembles with Active Constraints*, 2009, pp. 175–189.

[74] W. Xiao, Y. Yang, H. Wang, T. Li, and H. Xing, "Semi-supervised hierarchical clustering ensemble and its application," *Neurocomputing*, vol. 173, pp. 1362–1376, 2016.

[75] E. Dimitriadou, A. Weingessel, and K. Hornik, "A mixed ensemble approach for the semi-supervised problem," in *ICANN*, 2002, pp. 571–576.

[76] J. Kittler, "On combining classifiers," *PAMI*, vol. 20, no. 3, pp. 226–239, 1998.

[77] G. Forestier, P. Gançarski, and C. Wemmert, "Collaborative clustering with background knowledge," *Data Knowl Eng*, vol. 69, no. 2, pp. 211–228, 2010.

[78] C. Wemmert, P. Gançarski, and J. Korczak, "A collaborative approach to combine multiple learning methods," *Int J Artif Intell Tools*, vol. 9, no. 1, pp. 59–78, 2000.

[79] P. Gançarski and C. Wemmert, "Collaborative multi-step mono-level multi-strategy classification," *MTAP*, vol. 35, no. 1, pp. 1–27, 2007.

[80] W. Pedrycz, "Collaborative fuzzy clustering," *Pattern Recognit Lett*, vol. 23, no. 14, pp. 1675–1686, 2002.

[81] B. Babaki, T. Guns, and S. Nijssen, "Constrained clustering using column generation," in *CPAIOR*, 2014, pp. 438–454.

[82] B. Babaki. (2017) MIPKmeans. [Online]. Available: https://github.com/Behrouz-Babaki/MIPKmeans

[83] I. Davidson, S. Ravi, and L. Shamis, "A SAT-based framework for efficient constrained

clustering," in *SDM*, 2010, pp. 94–105.

[84] T.-B.-H. Dao, K.-C. Duong, and C. Vrain, "Constrained clustering by constraint programming," *Artificial Intelligence*, vol. 244, pp. 70–94, 2017.

[85] Z. Li, J. Liu, and X. Tang, "Constrained clustering via spectral regularization," in *CVPR*, 2009, pp. 421–428.

[86] M. Cucuringu, I. Koutis, S. Chawla, G. Miller, and R. Peng, "Simple and scalable constrained clustering: A generalized spectral method," in *AIStats*, 2016, pp. 445–454.

[87] M. Bilenko, S. Basu, and R. Mooney, "Integrating constraints and metric learning in semi-supervised clustering," in *ICML*, 2004, pp. 11–18.

[88] N. Shental, A. Bar-Hillel, T. Hertz, and D. Weinshall, "Computing gaussian mixture models with EM using equivalence constraints," in *NIPS*, 2013, pp. 465–472.

[89] C. Wemmert, "Classification hybride distribuée par collaboration de méthodes non supervisées," Ph.D. dissertation, Strasbourg 1, 2000.

[90] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, and J. Wortman Vaughan, "A theory of learning from different domains," *Mach Learn*, vol. 79, pp. 151–175, 2010.

[91] D. Tuia, C. Persello, and L. Bruzzone, "Domain adaptation for the classification of remote sensing data: An overview of recent advances," *IEEE Geosci Remote Sens Mag*, vol. 4, no. 2, pp. 41–57, 2016.

[92] R. Hadria, B. Duchemin, F. Baup, T. le Toan, A. Bouvet, G. Dedieu, and M. le Page, "Combined use of optical and radar satellite data for the detection of tillage and irrigation operations: Case study in central morocco," *Agric Water Manag*, vol. 96, no. 7, pp. 1120–1127, 2009.

[93] C. Marais Sicre, F. Baup, and R. Fieuzal, "Determination of the crop row orientations from formosat-2 multi-temporal and panchromatic images," *ISPRS J Photogramm Remote Sens*, vol. 94, pp. 127–142, 2014.

[94] B. Duchemin, R. Fieuzal, M. Rivera, J. Ezzahar, L. Jarlan, J. Rodriguez, O. Hagolle, and C. Watts, "Impact of sowing date on yield and water-use-efficiency of wheat analyzed through spatial modeling and formosat-2 images," *Remote Sens.*, vol. 7, pp. 5951–5979, 2015.

[95] L. Hubert and P. Arabie, "Comparing partitions," *Journal of classification*, pp. 193–218, 1985.

[96] K. Wagstaff, S. Basu, and I. Davidson, "When is constrained clustering beneficial, and

why?" in *IAAI*, 2006.

[97] I. Davidson, K. Wagstaff, and S. Basu, "Measuring constraint-set utility for partitional clustering algorithms," in *PKDD*, 2006, pp. 115–126.

[98] P. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, 1987.

[99] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," in *NIPS*, 2004.

[100] X. Zhang, L. Jiao, F. Liu, L. Bo, and M. Gong, "Spectral clustering ensemble applied to SAR image segmentation," *TGRS*, vol. 46, pp. 2126–2136, 2008.

Thomas Lampert received a B.Sc. (Hons.) in Computer Science and an M.Sc. (with distinction) in AI from the University of Exeter, U.K., and a Ph.D. in pattern recognition from the University of York, U.K., under the sponsorship of QinetiQ and DSTL He has held research positions at the University of York, CNRS, and University of Strasbourg.

He currently holds a postdoctoral position at ICube Laboratory. His research interests include the theoretical aspects of machine learning, and their application to remote-sensing and medical imaging.

Baptiste Lafabregue received a B.Sc. in computer science (2012) from the University of Strasbourg, France, and an M.Sc. in computer science (2014) from ENSIIE, France.

He is currently a PhD student in the SDC research team, ICube, and works in collaboration with SERTIT, a remote sensing technology and services platform. His research interests are primarily in unsupervised clustering, with a specific interest in remote sensing imagery and time-series.

Thi-Bich-Hanh Dao received a Ph.D. in computer science from the University of Aix-Marseille II (France) in 2000 and Habilitation in computer science from the University of Orléans (France) in 2018. She is currently Assistant Professor at the University of Orléans. Her current research topics include constraint programming, clustering and pattern mining.

Nicolas Serrette received a B.Sc. and an M.Sc. in computer science from the University of Orléans, France, in 2015 and 2017 respectively.

Christel Vrain is a full professor at the University of Orléans. She studied Mathematics in ENS Fontenay aux Roses, she received a Ph.D. degree in Computer Science from the University of Paris 11 (France) in 1987 and a HDR (Habilitation to lead research) from the University of Orléans (France) in 1996. Her current research interests include clustering, inductive logic programming, pattern mining, constraint programming and applications.

Pierre Gançarski received a Ph.D. and Habilitation in computer science from the University of Strasbourg, France, in 1989 and 2007, respectively.

He is currently a full professor of computer science at the University of Strasbourg. His current research interests include collaborative multistrategical clustering with applications to complex data mining, and remote sensing image and time-series analysis.
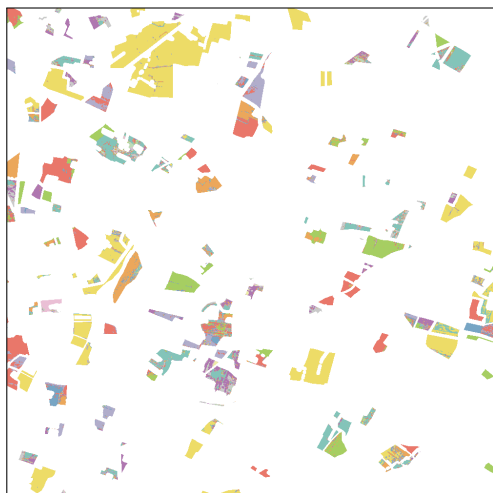
(a) DTW (classification ARI $= 0.507$, overall ARI $= 0.525$, training constraint fraction $= 15\%$)
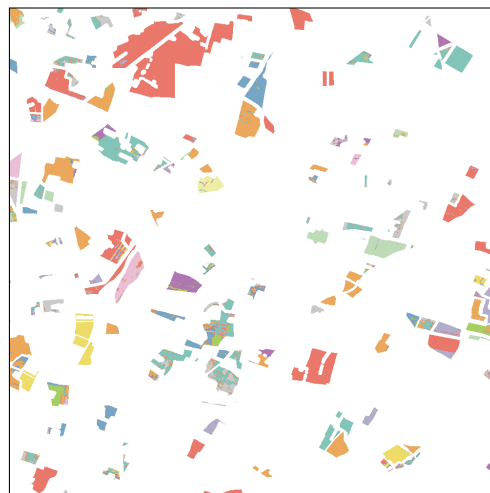
(b) Euclidean (classification ARI $= 0.522$, overall ARI $= 0.540$, training constraint fraction $= 5\%$)

(c) DTW clustering contingency table ($\chi^2 = 25231.030$, $p = 0.000$, clustering ARI $= 0.679$, constraint fraction $= 15\%$)

(d) Euclidean clustering contingency table ($\chi^2 = 34104.814$, $p = 0.000$, clustering ARI $= 0.713$, constraint fraction $= 5\%$)

(e) Image (f) Reference Data (g) COP-KMeans (h) MIP-KMeans (i) Spec (j) CCSR (k) SAMARAH (l) CPClustering
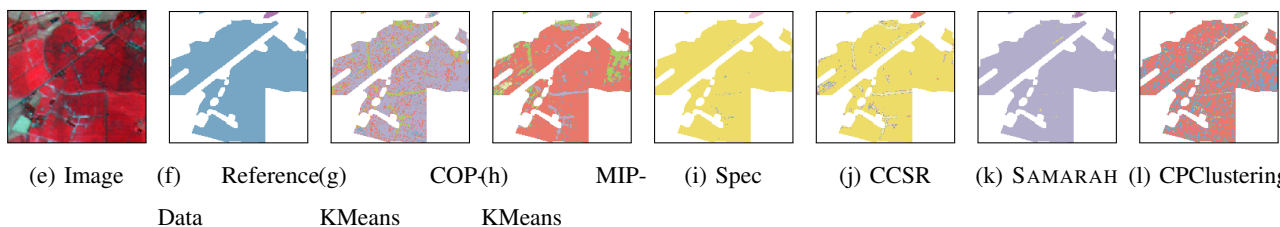
Figure 3. SAMARAH (a) & (b) out-of-sample classification (see text for details); (c) & (d) clustering contingency tables (rows have been normalised to emphasise the distribution of each class over the clusters); (e)–(l) subset of the image highlighting the noise present in the data and image classification using the maximum ARI clusterings (DTW distance measure).
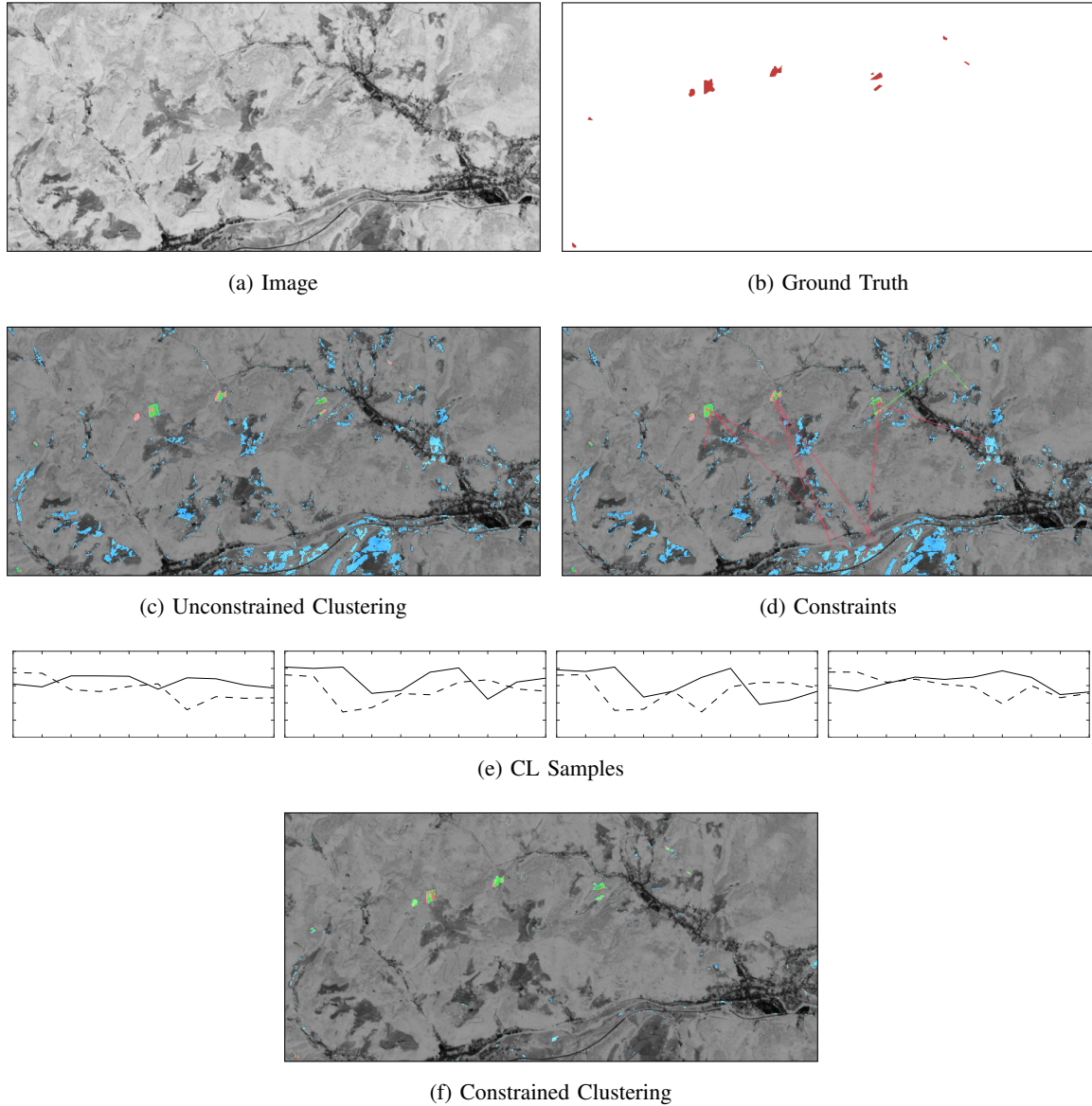
(a) Image

(b) Ground Truth

(c) Unconstrained Clustering

(d) Constraints

(e) CL Samples

(f) Constrained Clustering

Figure 4. (a) Clear-cuts NDVI time-series image (t_{10} displayed here), (b) tree cut ground truth locations, (c) DTW unconstrained clustering result (green - true positive, red - false negative, blue - false positive), (d) user selected constraints, (e) four of the sample pairs under CL constraints (the dashed lines are tree-cuts samples), and (f) DTW constrained clustering results. Images in (c), (d), (f) have been darkened to highlight the clustering results.
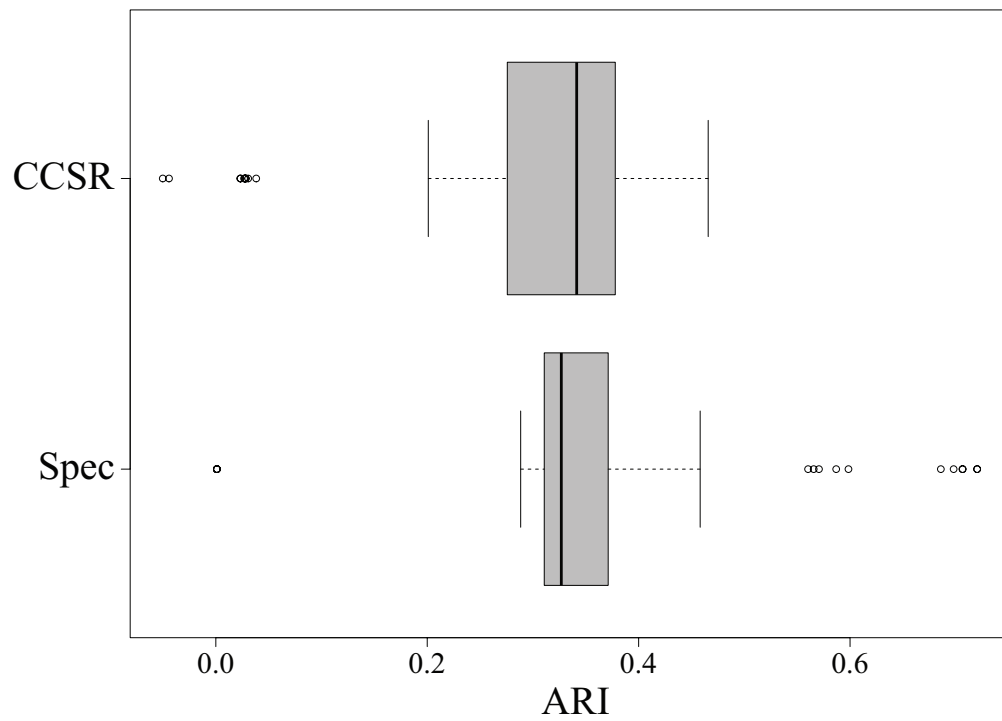
Figure 5. Distribution of spectral method training ARIs when different parameter values are chosen during the grid search. Spec — mean: 0.367, min: 0.001, max: 0.720, standard deviation: 0.184. CCSR — mean: 0.297, min: −0.050, max: 0.466, standard deviation: 0.139.