Semantic Clustering of a Sequence of Satellite Images

Carlos Echegoyen, Aritz Pérez, Guzmán Santafé, Unai Pérez-Goya and María Dolores Ugarte

***Abstract*—Satellite images constitute a highly valuable and abundant resource for many real world applications. However, the labeled data needed to train most machine learning models are scarce and difficult to obtain. In this context, the current work investigates a fully unsupervised methodology that, given a temporal sequence of satellite images, creates a partition of the ground according to its semantic properties and their evolution over time. The sequences of images are translated into a grid of multivariate time series of embedded tiles. The embedding and the partitional clustering of these sequences of tiles are constructed in two iterative steps: In the first step, the embedding is able to extract the information of the sequences of tiles based on a geographical neighborhood, and the tiles are grouped into clusters. In the second step, the embedding is refined by using the neighborhood defined by the clusters, and the final clustering of the sequences of tiles is obtained. We illustrate the methodology by conducting the semantic clustering of a sequence of 20 satellite images of the region of Navarra (Spain). The results show that the clustering of multivariate time series is robust and contains trustful spatio-temporal semantic information about the region under study. We unveil the close connection that exists between the geographic and embedded spaces, and find out that the semantic properties attributed to these kinds of embeddings are fully exploited and even enhanced by the proposed clustering of time series.**

***Index Terms*—Clustering, deep learning, machine learning, semantic embeddings, satellite images, time series, unsupervised learning.**

I. Introduction

Earth monitoring through the analysis of satellite images is nowadays essential in the identification, mapping, assessment, and monitoring of land use and cover changes. This land monitoring throughout long periods of time is possible and cost-effective thanks to multi-spectral satellite images freely provided by satellite programs supported by public agencies. Thus, public access to satellite imagery has favored the interest of a growing number of researchers in the analysis of satellite image time series (SITS). Additionally, the huge data volume and the complexity of the SITS analysis have promoted the use of machine learning methods. More specifically, supervised classification methods have been used, for instance, to obtain land use maps, land cover maps, crop classification or harvest prediction [1], [2], [3]. Although the access to data from satellite imagery is not a limitation, obtaining labeled data for supervised classification methods may be problematic since these kinds of data are very expensive to produce and maintain. Therefore, semi-supervised and clustering methods are gaining more and more relevance in SITS analysis [4], [5], [6], [7], [8].

SITS data have been originally exploited at pixel level [9], [10], [11]. In these approaches, series of pixels corresponding to the same geographical position throughout a temporal sequence of satellite images are further compared to each other and associated to different classes or clusters. However, in recent years, the rapid development of deep learning techniques, and more specifically convolutional neural networks (CNN), represent a revolution in the field of image analysis in general and in the analysis of SITS data in particular [12], [13]. These kinds of techniques are able to extract patterns and insights from vast amounts of complex data. Therefore, they become a natural candidate to solve problems in the field of remote sensing, where the data coming from different satellites is growing dramatically.

Currently, the application of deep learning techniques cover a wide area, ranging from predicting sea ice motion [14] to wildfire forecasting [15], land use classification [16] and crop type mapping that combines data from satellites and farmer smartphones [17], to name a few. In the case of SITSs, deep learning techniques have also become a useful tool [13], [18]. Taking advantage of the temporal dimension, they provide a more reliable solution to a wide range of problems [19]. Additionally, we consider that the explicit use of time series also open up a wide variety of possibilities for change-point detection in remote sensing data.

In the analysis of image data, an autoencoder is a type of CNN used for unsupervised dimensionality reduction or feature learning. In remote sensing, autoencoders have been used to create embeddings which are able to extract features from satellite imagery before using other classification or clustering methods [20], [7], [21]. However, the feature extraction obtained by an autoencoder embedding is guided by a good compression of the information reported in the original image, and it is not directly related to a classification or clustering purpose. With the aim of creating semantically meaningful embeddings, algorithms such as Tile2Vec [6] have been developed recently. In particular, this algorithm is learned in an unsupervised way by means of CNNs and tries to create an embedding in which similar tiles have vector representations which are close to each other and distant to different tiles. The learning of the model is based on triplets of tiles. The objective is to find the parameters that minimize the distance between geographically neighboring tiles and maximize the

This work has been supported by Project PID2020-113125RB-I00/MCIN/AEI/10.130 39/501100011033. Carlos Echegoyen, Guzmán Santafé, Unai Pérez-Goya and María Dolores Ugarte are members the Spatial Statistics Goup, Public University of Navarre, 31006 Pamplona, Spain. Email: {carlos.echegoyen, guzman.santafe, unai.perez, lola.ugarte}@unavarra.es

Aritz Pérez is at the Basque Center for Applied Mathematics. Email: aperez@bcamath.org

distance between distant tiles. This embedding is analogous to the well known Word2Vec [22], but instead of encoding words, Tile2Vec encodes tiles of the satellite image.

Taking into account the aforementioned elements, the methodology presented in this work analyzes SITSs in a completely unsupervised manner by clustering together areas of the image that represent similar geographical characteristics and also similar evolution in the time series. The procedure can be summarized as follows. Firstly, we train the embedding based on Tile2Vec according to the geographic neighborhood in the context of SITS. Secondly, the images are decomposed into grids of tiles so that the semantic embedding is used to obtain the embedded representation of each tile. Accordingly, the sequence of images is represented as a collection of multidimensional time series (MTS) corresponding to each sequence of tiles (ST). Thirdly, we cluster the areas with similar behavior over time using an adaptation of the K-means clustering method for MTS. The use of the K-means clustering is motivated by the relation between Tile2Vec's loss function and the K-Means' loss function. Fourthly, we extend the training of the embedding with new data, but this time considering the neighborhood given by the clustering partition of MTS. Finally, we run a second clustering based on the refined embedding.

The resultant partition is analyzed in different ways. Firstly, we plot the obtained clustering partition over the satellite images by assigning a color to every sequence of tiles belonging to the same cluster. The color given to each cluster is related to the semantic information encoded by its centroid. Therefore, similarity among partition colors is related to semantic similarity among them. Secondly, we visualize the clustering of the MTS through two-dimensional projections, and compare the clustering in the geographic space and in the embedded space. Finally, we conduct a visual inspection of the semantics for each cluster by looking for the closest MTS to each centroid and also by exploring representative images close to the path between centroids of different clusters.

The results show that this procedure creates semantically reliable land partitions that are able to go beyond the superficial details of the image. This semantic arrangement is distinguished by finding structured patterns in the image where the clusters tend to cover wide and compact areas that put together mountains, crops, hills, riverbanks and other semantically related elements that evolve similarly over time without the restriction of predefined labels. The study has been carried out by using images from the region of Navarre (Spain) considering the four seasons of the year during the last 5 years. This case study represents a real scenario, where any given region could be the subject of study. The management of the satellite images has been assisted by the `rsat` package [23].

The rest of the paper is organized as follows. Section II discusses relevant previous works. Section III summarizes the background of the Tile2Vec embedding. Section IV develops the proposed methodology based on a semantic embedding, and partitional clustering. Section V specifies the details of the experiments and the model parameters used for the current research. In Section VI the empirical results of the study are presented and discussed. Finally, Section VII draws the conclusion obtained during the study and points out possible future work.

II. Related work

An important drawback when dealing with satellite images by means of machine learning methods is the need for labeled data [24]. Most of the works have been focused on supervised classification techniques. However, due to the difficulty of obtaining the ground truth when dealing with satellite images, an increasing number of contributions are devoted to develop unsupervised methods. For instance, [25] develops a procedure based on dynamic time wrapping (DTW) distance measures, [26] carries out unsupervised learning of SITS by adapting a constrained k-means clustering algorithm and using DTW distance measure, and [27] uses a graph-based strategy to represent the temporal evolution of specific areas in the image and clusters this evolution graphs to identify spatio-temporal entities that evolve similarly. However, this last approach relies on a segmentation phase to extract the spatial entities to track over time.

In parallel to the above, the creation of embedding by means of deep learning techniques [28], [6], [29], [30], [31] is gaining increasing attention in the field of satellite images, among other things, due to the complexity of multi-spectral satellite images analysis. These methods are able to create embedded spaces where the images are encoded as vectors of a bounded size. Among these models, we are particularly interested in those that are able to create semantic embeddings where not only the vectors are meaningful but also the distances between them represent the reality of the images.

To the best of our knowledge, this kind of semantic embedding has not been directly incorporated and analyzed within the context of clustering of SITS. Although in [7] the authors aim at clustering spatial areas with similar temporal evolution, they use 3D convolutional autoencoders where the whole TSs are compressed as a vector and the distance between these vectors has no special meaning. Therefore, the information contained in the TS and the relationship among them are obscured by the encoding given by the autoencoder, which provides a good compression of the image, but it does not encourage vector discrimination.

In [32], the authors present a supervised learning method to train CNNs that improves class discrimination in scene classification. They modify the objective function of CNNs with a new proposed function so that in the feature space obtained by the CNN, images from the same scene class are mapped close to each other and images of different classes are mapped as farther apart as possible. This idea is transferred to unsupervised learning by means of semantic embeddings. As introduced before, Tile2Vec [6] uses a CNN to project the image's tiles into in an embedded or latent space, so that, similar tiles are mapped close to each other and different tiles are mapped as farther apart as possible. Similarly, [33] presents an analogous approach, but using hexagonal instead of squared tiles. Alternatively, [34] uses a SimCLR approach, an encoder network trained to maximize agreement by using contrastive loss [35], and modifies this model to include multiple neighbor

tiles. Thus, k-neighbor tiles are used and no distant tile is taken into consideration.

The use of Til2Vec in the current paper is mainly motivated by the empirical demonstration that the authors provide in [6] regarding the properties of the embedded space. The representation given by Til2Vec is semantically meaningful and moreover, simple arithmetic operations within the space conserve semantic properties. This provides us with a solid basis for analyzing the final clustering partition obtained by the proposed methodology from a semantic perspective.

Finally, it is worth mentioning that the clustering and classification of SITS is closely related with the automatic generation of land cover maps, which has experienced a rapid development during the last decade [36], [37] due to the increasing availability and quality of satellite imagery data. Also in this more specific context, the development of innovative methods based on deep learning techniques opens up new research opportunities [24], [7], [38].

III. Tile2Vec in a nutshell

A well-known successful semantic embedding is Word2Vec [39], [22], which has been widely applied to solve natural language processing problems. This word embedding is based on the distributional hypothesis, i.e., words that appear in the same context tend to have similar meanings. When this is translated to static satellite images, the context is given by the spatial neighborhood, as stated by the Tobler's first law of geography [40]: *everything is related to everything else, but near things are more related than distant things*. This idea is succinctly put into practice by Tile2Vec [6]. As atomic units, this algorithm considers tiles x of fixed dimensions as image patches taken from a satellite image X. Following Tobler's law, the learning algorithm of the embedding assumes that, on average, closer tiles are more similar than distant tiles, and therefore, their embedded representation has to be closer. The learning process is expected to build not only an embedded space where vectors of similar images are closer than vectors of dissimilar images, but also to capture the corresponding degree of similarity.

Tile2Vec is learned from a training set of triplets of tiles (x_a, x_b, x_c), where x_a denotes the anchor tile, x_b the neighbor tile and x_c the distant tile. The embedding function is given by a ResNet-18 [41] architecture with a modified input, to be able to handle multi-spectral tiles, and without the final classification layer. The embedding function f maps a tile $x \in \mathcal{X}$ to a d-dimensional vector z, $f : \mathcal{X} \rightarrow \mathbb{R}^d$, where $\mathcal{X}$ is the domain of tiles, and it is found by minimizing the following loss function:

$$L(D) = \sum_{(x_a,x_b,x_c)\in D} [||f(x_a) - f(x_b)||_2 - ||f(x_a) - f(x_c)||_2 + \delta]_+, \quad (1)$$

where $D = \{(x_a, x_b, x_c)\}$ is the training set of triplets, $\delta \geq 0$ is the margin, and $[\cdot]_+$ is the positive part of the argument. In a nutshell, the learning algorithm finds the embedding function f that minimizes the Euclidean distance between an anchor and its neighbor while maximizing the Euclidean distance between the anchor and the distant tile over the tile triplets in the training set. For further details, see [6].

In [6], the authors show that the Tile2Vec embedding is able to successfully extract semantic information from a set of tiles. They create interpolations and analogies, e.g., between field and urban tiles, and include several experiments that show robust results with different configurations, datasets, and problems. The semantic of these kinds of embeddings has been further explored by creating analogies and compositions in the embedded space with algebraic operations such as addition and difference [22], [42] or by learning more complex operations [43].

We argue that Tile2Vec emerges as a natural candidate to build a *multidimensional time series* (*MTS*) *embedding for sequences of tiles* (*ST*) on which performing partitional clustering makes sense.

IV. Clustering of the embedding of STs

In this work, we propose a methodology for constructing an embedding of STs given in terms of MTSs. The embedding allows us to perform spatio-temporal clustering of a sequence of satellite images, grouping regions that exhibit similar evolving patterns. We assume that the images can be decomposed into tiles that contain relevant geographic and temporal information when considered in isolation. Thus, the size of the tile should be the minimum that allows an expert to determine relevant spatial and temporal characteristics of the region. For instance, tiles in isolation should allow identifying semantic entities such as rivers, mountains, hills, crops or pastures. The embedding is first constructed by extracting spatial information according to the distributional hypothesis given by Tobbler's 1^{st} law of geography, and then it is refined by using clustering information.

Let $X = \{x_1, ..., x_m\}$ be an image of a region that is decomposed into a grid of tiles x_i of size m for $i = 1...m$. Let $(X^1, ..., X^T)$ be a temporal sequence of satellite images of the same region, where X^t is the image at time t, for $t = 1, ..., T$. From these images, we get sequences of tiles $\boldsymbol{X} = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_m\}$, where $\boldsymbol{x}_i = (x_i^1, ..., x_i^T)$ corresponds to the i-th ST. Based on Tile2Vec, we represent STs as MTSs of embedded vectors. Overall, we propose the following procedure to perform the semantic clustering:

- Learn a geographic-based embedding of STs, f^g (Subsection IV-A).
- Clustering of STs using the embedding f^g (Subsection IV-B).
- Learn a clustering-based embedding of STs, f^c (Subsection IV-C).
- Clustering of STs using the embedding f^c (Subsection IV-C).

The procedure starts by learning the embedding f^g using a training set of triplets taken from the images $\{X^1, ..., X^T\}$, where each triplet belongs to the same time and, neighbor and distant tiles are defined according to a spatial distance. Using the embedding function, f^g, we represent the ST $\boldsymbol{x}_i$ as a MTS, $\boldsymbol{z}_i^g = (f^g(x_i^1), ..., f^g(x_i^T))$, for $\boldsymbol{x}_i \in \boldsymbol{X}$. The embedding of a sequence of images using a generic Tile2Vec embedding function f is illustrated in Figure 1. Secondly, we construct a partitional clustering of the embedded grid of STs

$\boldsymbol{Z}^g = \{\boldsymbol{z}_1^g, ..., \boldsymbol{z}_m^g\}$, $\mathcal{P}^g = \{P_1^g, ..., P_K^g\}$ with $P_k^g \subset \boldsymbol{Z}^g$ for $k = 1, ..., K$. Thirdly, we learn an embedding f^c learned from triplets obtained from the embedded grid of STs $\boldsymbol{Z}^g$, where each triplet belongs to the same image, and neighbor and distant tiles correspond to embedded tiles from the same and different clusters, respectively. The clustering-based embedding f^c constitutes a refinement of the geographic-based embedding f^g, which captures the spatio-temporal patterns that characterize the identified clusters $\mathcal{P}^g$. Using the function f^c, we embed again each ST $\boldsymbol{x}_i$ as $\boldsymbol{z}_i^c = (f^c(x_i^1), ..., f^c(x_i^T))$, for $i = 1, ..., m$. Lastly, the final clustering of STs is obtained in the embedded space given by f^c. In the remainder of this section, we provide a detailed explanation of our proposal.

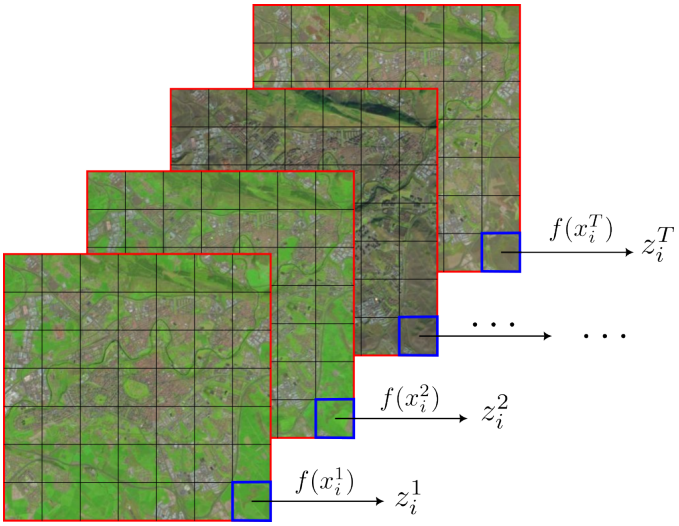


Fig. 1. Encoding of a sequence of tiles (ST) $\boldsymbol{x}_i$ by means of the embedding f, $\boldsymbol{z}_i = (f(x_i^1), f(x_i^2), ..., f(x_i^T))$.

A. Geographic-based embedding of STs

We aim at constructing an embedding for STs, therefore, we have to adapt the training of the Tile2Vec model to this context. Since we want to capture the semantic of a region as a whole for any given time, we generate the training set D^g of tile triplets by using sequences of images. To create a triplet, we consider two sequences of images $(X^1, ..., X^T)$ and $(Y^1, ..., Y^T)$ with the same timestamps subject to the next constraint: the triplet must belong to images from the same time, (x_a^t, x_b^t, y_c^t) where $x_a^t, x_b^t \in X^t$ and $y_c^t \in Y^t$. Due to this temporal constraint, intuitively, our embedding of tiles will be focused on the extraction of semantic information based on the spatial component of every image conforming the sequence $(X^1, ..., X^T)$. As shown above, the sequence of images $(Y^1, ..., Y^T)$ is used to obtain distant tiles. Thus, by considering the use of distant tiles from another sequence of images, we obtain a richer geographic-based embedding.

The training set D^g consists of N triplets, each of one generated following the next process:

- Select t uniformly at random from $\{1, ..., T\}$
- Select an anchor x_a^t uniformly at random from the image X^t
- Select a neighbor x_b^t uniformly at random from a ball of radius r of X^t centered at x_a^t
- Select a distant tile y_c^t uniformly at random from Y^t corresponding to the same timestamp from the sequence $\boldsymbol{Y}$

The process of the generation of the training set of triplets D^g is illustrated in Figure 2.

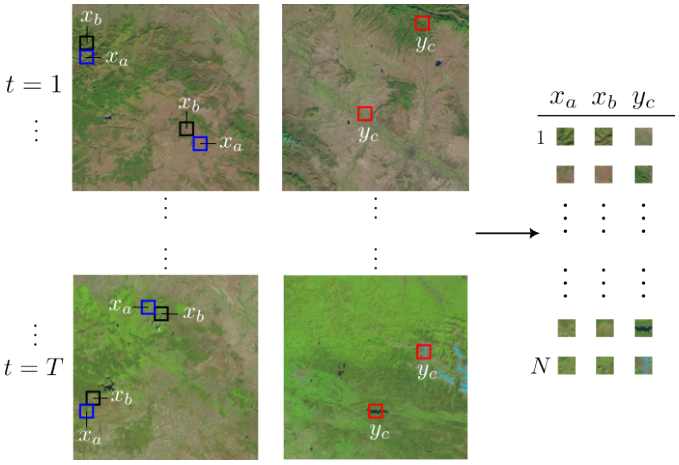


Fig. 2. Scheme to illustrate the generation of the dataset of tile triplets.

Once the dataset of triplets D^g is created, we learn the geographic-based embedding function f^g from D^g by minimizing Equation 1. The embedding function f^g maps the space of the tiles $\mathcal{X}$ into $\mathbb{R}^d$, where d is the dimension of the embedding.

B. Clustering STs

Given the embedding of a grid of sequences of tiles, $\boldsymbol{Z} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_m\}$, we would like to identify K distinct groups of sequence of tiles by using partitional clustering techniques. In particular, we propose to solve the K-means problem for $\boldsymbol{Z}$, where K determines the number of subgroups (clusters) of a partition of $\boldsymbol{Z}$ (clustering), $\mathcal{P} = \{P_1, ..., P_K\}$, with non-empty clusters $P_k \subset \boldsymbol{Z}$ for $k = 1, ..., K$. The K-means problem consists of finding a clustering $\mathcal{P}$ that minimizes the error:

$$E(\mathcal{P}) = \sum_{k=1}^{K} \sum_{\boldsymbol{z} \in P_k} d(\boldsymbol{z}_k, \boldsymbol{c}_k)^2, \tag{2}$$

where $d(\boldsymbol{z}, \boldsymbol{z}') = \sum_{t=1}^{T} ||z_t - z'_t||_2$ is the Euclidean distance between the MTSs $\boldsymbol{z}$ and $\boldsymbol{z}'$, and $\boldsymbol{c}_k = \frac{1}{|P_k|} \sum_{\boldsymbol{z} \in P_k} \boldsymbol{z}$ is the centroid of the cluster P_k which corresponds to the average of the MTS within this cluster. The K-means problem is NP-hard, and the Lloyds's algorithm [44] (a.k.a. the K-means algorithm) is used to obtain a solution. The Lloyd's algorithm has been identified as one of the top 10 algorithm in data mining [45].

The Lloyd's algorithm is an iterative procedure that generates a sequence of clusterings with a monotone decreasing error function (Eq. 2), until its convergence to a fixed point. The algorithm is linear in the number of considered data points, m, and therefore, the proposed methodology can be

applied to large images. In Appendix A we show the link between an adaptation of the Tile2Vec for partitional clustering and the K-means error of the sequence of clusterings obtained by the Lloyd's algorithm.

In [6], the authors show that the interpolation of two tiles using the Tile2Vec embedding allows covering the full spectrum of intermediate patterns. The Lloyd's algorithm obtains convex clustering[1]. Convex clusterings are particularly interesting from the Tile2Vec embedding perspective because the convex combination of any subset of points from a cluster $P_k \in \mathcal{P}$ belongs to the convex hull of P_k. In other words, both the interpolation of any two points and the centroid of a cluster P_k belong to the class of points given by the cluster P_k.

C. Clustering-based embedding of STs

The last step consists of refining the geographical-based embedding f^g by using information obtained from the partitional clustering $\mathcal{P}^g$ and conducting the final clustering of STs using the new embedding f^c. For this purpose, we generate a training set of triplets D^c using a neighborhood based on $\mathcal{P}^g$. With a slight abuse in the notation, in this section, we consider that the clustering $\mathcal{P}^g$ corresponds to the partition of the sequences of images $\{X^1, ..., X^T\}$ associated to the geographic-based embedding $\boldsymbol{Z}^g$. The triplets conforming D^c, (x_a^t, x_b^t, x_c^t), again satisfy the temporal constraint. The training set D^c consists of M triplets, where each of them is generated following the next process:

- Select a cluster index k at random from $\{1, ..., K\}$ with a probability proportional to the size of the cluster $|P_k^g|$.
- Select an anchor ST $\boldsymbol{x}_a$ uniformly at random from the cluster P_k^g.
- Select a neighbor ST $\boldsymbol{x}_b \neq \boldsymbol{x}_a$ uniformly at random from the cluster P_k^g.
- Select a cluster index j at random from $\{1, ..., k-1, k+1, ..., K\}$ with a probability proportional to the size of the cluster $|P_j^g|$.
- Select a distant ST $\boldsymbol{x}_c^t$ uniformly at random from P_j^g.
- Construct the triplet (x_a^t, x_b^t, x_c^t) by selecting t uniformly at random from $\{1, ..., T\}$.

Now, we learn the clustering-based embedding f^c by extending the training of f^g using the new training set D^c. The obtained clustering-based embedding f^c, is still a mapping from the tile space $\mathcal{X}$ into $\mathbb{R}^d$. The cluster based embedding will decrease the average intra-cluster dissimilarity of $\mathcal{P}^c$ with respect to the geographical-based embedding, while increasing the average inter-cluster dissimilarity.

Finally, the STs are re-clustered using the Lloyd's algorithm over $\boldsymbol{Z}^c = \{(f^c(x_i)^1, ..., f^c(x_i)^T) : \text{ for } i = 1, ..., m\}$ (see Section IV-B). This last clustering is a refinement of the clustering obtained for $\boldsymbol{Z}^g$, where the clustering $\mathcal{P}^c$ tends to show a better separation between its conforming clusters.

[1]Convex clustering: We say that a partitional clustering $\mathcal{P}$ is convex when the convex hulls of its corresponding clusters $P \in \mathcal{P}$ are pairwise disjoint

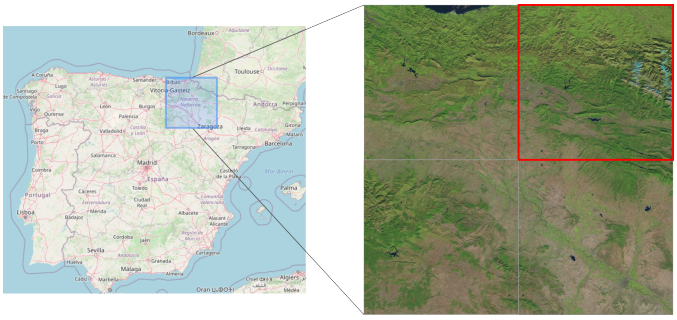


Fig. 3. Region of Navarre in northern Spain. The training is conducted with the whole area, while the clustering is focused on the area marked in red.

V. Experimental design

In this section, we illustrate our proposal by using a sequence of Sentinel-2 images from the region of Navarre (northern Spain). Firstly, we provide the details of the learning parameters and the satellite imagery dataset. Secondly, we explain the three kinds of results we use to analyze the proposed methodology and the region of interest.

A. Image dataset and training parameters

We use Sentinel-2 RGB bands to create images of size 10980×10980 with spatial resolution of 10 meters per pixel. These three bands, along with the near infrared, are the only bands provided by Sentinel-2 with this resolution. The remaining bands are given at 20m and 60m. Since this work stresses the use of MTSs and part of the experiments rely on visual inspection, we choose to deal with images of bounded complexity in terms of band composition to facilitate the interpretation of the results.

The region selected for the current research is the area surrounding the province of Navarre in Spain (see Figure 3). This area contains a variety of land types such as mountains or crops, and it exhibits different characteristics along the year, such as snow-covered areas or harvested fields. The selection of this area demonstrates that the methodology presented in this paper works in practice. We emphasize that our proposal is general and can be applied to any other places, resolutions and bands.

The whole area considered for embedding training is covered by 4 satellite images (see Figure 3 on the right). Note that this scheme can be extrapolated directly to any other region of interest. To maintain a balance between complexity and soundness of the results, the final analysis focuses on the sequence of images, $(X^1, ..., X^T)$, corresponding to the area marked in red in Figure 3. The other three areas will be used to conform $(Y^1, ..., Y^T)$, the sequence of images from which the distant tiles are sampled. We get images of each season of the year during the last five years (2017-2021). Therefore, we use a total of 4 (regions) $\times 5$ (years) $\times 4$ (seasons) $= 60$ Sentinel-2 images to train the embedding f^g.

We firstly train the geographic-based embedding, f^g, following the procedure proposed in Section IV-A and the experiments carried out in [6] are taken into consideration to fix the values of the learning parameters. Thus, $N = 100000$

triplets sampled from the Sentinel-2 images, 5000 triplets from each timestamp. The size of the tiles is 100×100 pixels (covering 1 km^2), the geographical neighborhood is given by a ball of radius $r = 50$, and the distant tile is always chosen from a different region with the same timestamp to amplify the differences between neighbors and distant tiles. The training process is iterated 50 epochs, with a batch size of 50 and a margin of $\delta = 50$. The last layer of the network has $d = 512$ features, which correspond to the number of dimensions of the embedding space for the tiles. For the clustering-based embedding f^c, we continue the learning of the model using the procedure described in Section IV-C, with $M = 20000$ triplets taken from $(X^1, ..., X^T)$ (the red region). The neighborhood is given by a partitional clustering of size $K = 5$. The training process is iterated 25 epochs, with a batch size of 50 and a margin of $\delta = 50$. We reduce the number of triplets and epochs due to the more specific nature of this refinement.

The experiments have a twofold purpose. Firstly, we study the combination of the following three basic elements: i) the semantic embedding whose training is guided by a triplet loss function, ii) the Lloyd's algorithm to conduct unsupervised learning within the embedded space and to establish the neighborhood of the second phase of training, and iii) the explicit use of MTS to create a rich, flexible and scalable framework. Secondly, we empirically show the semantically meaningful results obtained with the MTS embedding and the clustering.

B. Methods of analysis

a) Geographic representation of the embedded spaces: Given a clustering of the MTSs obtained with the Lloyd's algorithm, $\mathcal{P} = \{P_1, ..., P_K\}$, we plot the semantic representation as an image of the same spatial dimensions of the original satellite images. We assign the same color to the tiles belonging to the same cluster P_k for $k = 1, ..., K$. The colors are generated using principal component analysis (PCA). For this purpose, we use a PCA projection of the embedded STs, $\boldsymbol{Z}$, and the RGB colors are given by the first three PCA components. Specifically, a cluster P_k is represented by its centroid, $\boldsymbol{c}_k$. The color of the cluster is given by the first three components of the PCA projection for $\boldsymbol{c}_k$, for $k = 1, ..., K$. The centroid captures the overall semantic of the cluster. Due to the properties of the constructed embedding, the difference between the colors of a pair of clusters indicate their semantic similarity, which facilitates the interpretation of the obtained clustering. Additionally, we plot the semantic representation of the temporal sequence of images by using the colors given by the PCA for the embedding of each ST in the grid, $\boldsymbol{Z}$, without clustering. This kind of result provides a visual tool to gain intuition about the general spatio-temporal pattern of the region and, in particular, about the possible number of clusters behind the images. When we show the geographic representation of the clustering $\mathcal{P}^c$, we keep the same PCA colors as for $\mathcal{P}^g$ to ease the comparison of the generated images.

b) Projections of the embedded spaces: As a complement of the previous geographic representation, we show the clustering through a two-dimensional projection of the embedded space. The color of the clusters are the same as in the geographic representation, in order to study both representations together. Specifically, the original embedded space is projected down to two dimensions by using the t-Distributed Stochastic Neighbor Embedding (t-SNE) [46] and the Multidimensional Scaling (MDS) [47]. These methods are based on distances between points. We consider the euclidean distance between MTS. Then, each point corresponding to an MTS is depicted with the color of the cluster to which it belongs. The t-SNE is a probabilistic approach for manifold learning. It is well suited for the current research since it focuses on the local structure of the data and will tend to extract clustered local groups. In addition, this algorithm is used in previous works [6], [30] for the visualization of embeddings of tiles. On the other hand, MDS seeks a low-dimensional representation of the data that preserves the relative distances of the high-dimensional embedded space. The same representations are used for both, the geographical-based and clustering-based embeddings.

c) Interpolation of centroids: We illustrate the semantic behavior of the obtained cluster-based embedding by using the interpolation of the representation of pairs of STs. Specifically, we analyze the interpolation of pair of centroids. For this purpose, we take two centroids $\boldsymbol{c}_k$ and $\boldsymbol{c}_{k'}$, and we get intermediate MTS embeddings $\boldsymbol{z}_w = w \cdot \boldsymbol{c}_k + (1-w) \cdot \boldsymbol{c}_{k'}$, for $w \in [0, 1]$. Then, we get the three STs, $\boldsymbol{x}_k$, $\boldsymbol{x}_{k'}$ and $\boldsymbol{x}_w$ from $(X^1, ..., X^T)$, whose embeddings are the closest to $\boldsymbol{c}_k$, $\boldsymbol{c}_{k'}$ and $\boldsymbol{z}_w$, respectively. A particularly interesting interpolation corresponds to centroids from adjacent clusterings in the embedded spaces with $w = 0.5$, because it corresponds to an ST that falls in the boundary between the two clusters. These experiments show that the semantic properties revealed in [6] when dealing with isolated images, can be extrapolated to more complex contexts involving STs.

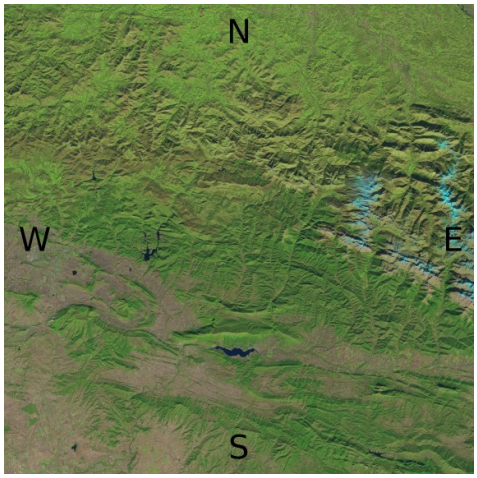
VI. Results

This section presents the results of the aforementioned experiments. Firstly, we show how the clustering arranges the STs both, in the geographic space and in the embedded space. Secondly, we study the impact that the clustering-based embedding has in the underlying structure of the clustering and in the corresponding external geographic representation. Lastly, we carry out a visual inspection of the semantic captured by the centroids and the interpolations between them.

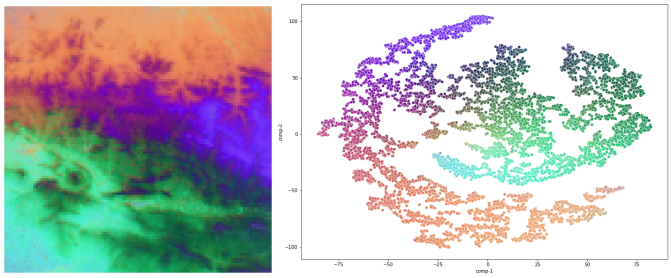
A. Geographic and embedded representations

A general overview of the region under study is presented in Figure 4. The terrain is illustrated in Figure 4(a) with one of the 20 images of the sequence. The mountains in the middle of the images correspond to the Pyrenees, where we can see snowy mountains on the east side.

Figures 4(b) and (c) show the geographic representation and the projection of the embedding, respectively. These images are generated by assigning a different color to each MTS embedding of STs according to PCA, as explained before. These images contain spatio-temporal information regarding



(a)

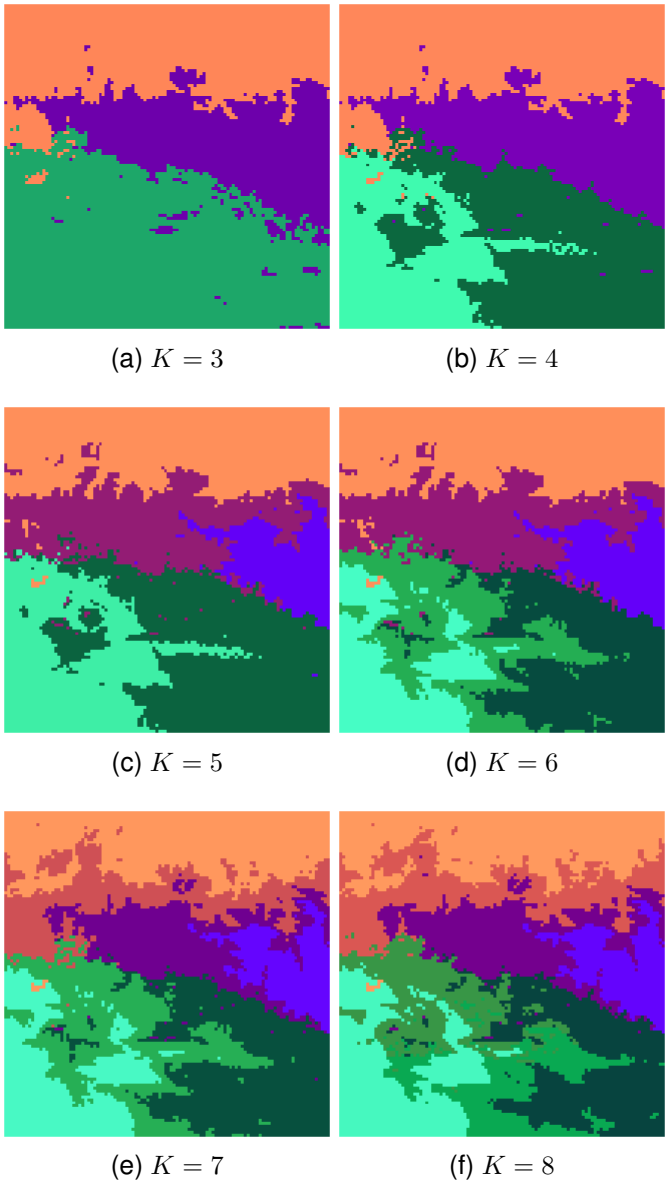


(b) (c)

Fig. 4. Results prior to the clustering with the embedding f^g: (a) Example of a satellite images belonging to the sequence for the region under analysis. (b) Geographic representation by assigning a different color for each ST according to PCA. (c) Projection of the embedded space.

the changing semantic of the different areas and the relationships among them. According to the colors, Figure 4(b) clearly shows three distinct big areas: the northern part after the Pyrenees, the southern part and the Pyrenees themselves. Although the comparison between Figures 4(a) and (b) is difficult since we have a single image on the one hand and a representation of 20 images on the other, we can see a strong connection between them. For instance, by inspection of the whole sequence of images (not shown here), we can check that the most intense violet-blue colors correspond to the area of the Pyrenees, where it snows frequently. In Figure 4(c) two main conglomerates can be observed, where the points corresponding to the Pyrenees have been arranged together with those belonging to the north. With this kind of preliminary analysis, it is possible to extract some useful information about the general patterns of the region, where some clear groups have emerged.

Figures 5 and 6 show the results of the clustering, $\mathcal{P}^g$, over the geographic-based embedding, f^g, with different number of clusters, $K \in \{3, 4, 5, 6, 7, 8\}$. We can observe that a hierarchical pattern naturally emerges in Figure 5. The big areas are mostly kept intact, and they are subdivided as the number of clusters increases, revealing additional details each time. In general, we can observe a very structured pattern, grouping large areas compactly. This suggests that the method



(a) $K = 3$ (b) $K = 4$

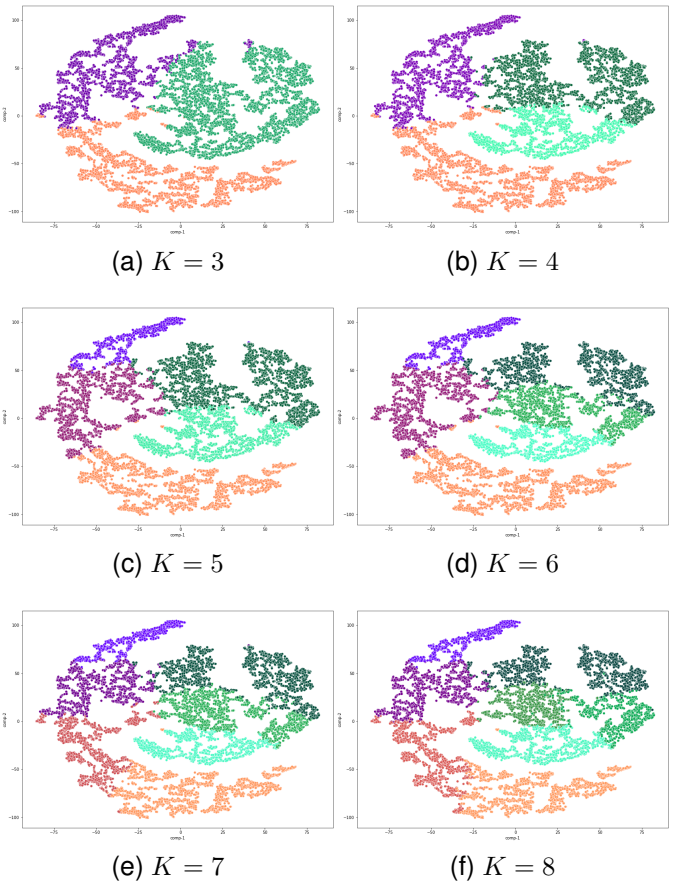
(c) $K = 5$ (d) $K = 6$

(e) $K = 7$ (f) $K = 8$

Fig. 5. Geographical representation of the clustering $\mathcal{P}^g$ as the number of clusters K increases

is not only capable of abstracting from the specific details of the STs, but also it is able to capture fine-grain semantics when higher numbers of clusters are allowed. The corresponding projection of the MTS embedded of different clusterings are presented in Figure 6. Similarly, it is clearly seen how well-defined groups appear as the number of clusters increases. Note that if two clusters are neighbors in the geographic representation, they are also neighbors in the projection and vice versa.

In particular, we can see in Figure 5 (c) that the southern region and the Pyrenees have been divided into two partitions with $K = 5$. We select this number of clusters to train the clustering-based embedding, f^c, because it entails a reasonable balance between richness of details and interpretability. Of course, the number of clusters can be set according to any requirement of the application at hand.



(a) $K = 3$ (b) $K = 4$

(c) $K = 5$ (d) $K = 6$

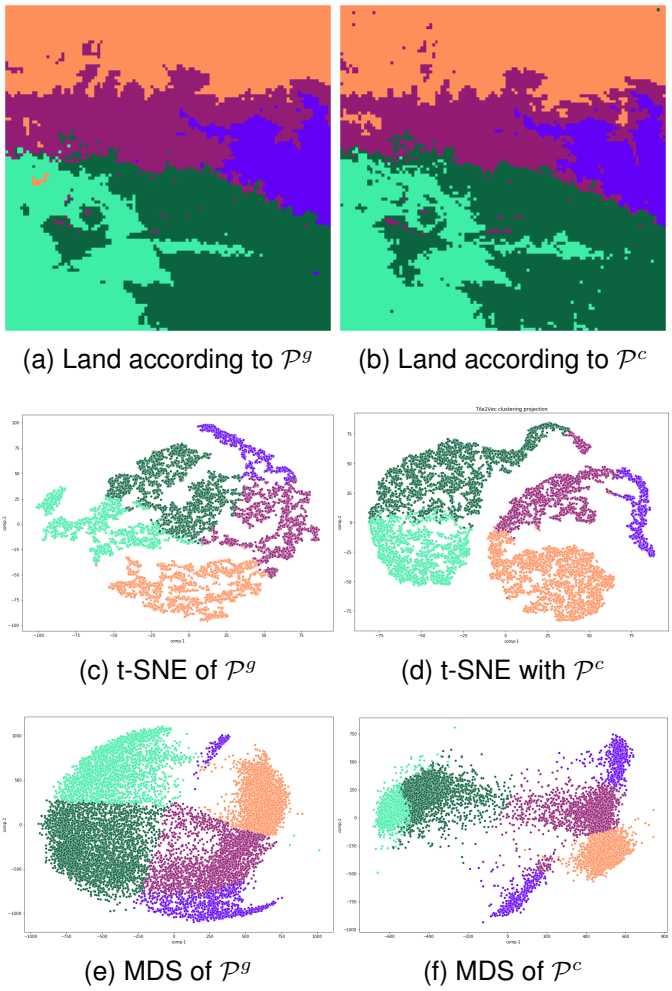
(e) $K = 7$ (f) $K = 8$

Fig. 6. t-SNE projection of the clustering $\mathcal{P}^g$ in the embedded space as the number of clusters K increases

B. The clustering-based embedding

This section presents the results of the clustering, $\mathcal{P}^g$, obtained with the clustering-based embedding, f^c, in comparison with those of the clustering, $\mathcal{P}^c$, obtained with the geographic-based embedding, f^g. For the sake of clarity, we plot the results with the same colors as in the previous figures.

In Figure 7, we can see that the geographic representations of $\mathcal{P}^g$ and $\mathcal{P}^c$ are almost the same, with small variations in some isolated tiles. This result suggests a convergence in the geographic representation of the semantic clustering. However, the rest of the charts reveal a dramatic change in the internal structure of the embedding. In this case, we use the two aforementioned projections: t-SNE (Figures 7 (c) and 7 (d)) and MDS (Figures 7 (e) and 7 (f)). Since t-SNE is probabilistic, the final result may vary slightly from run to run. We provide a second example of a t-SNE projection for $K = 5$ in Figure 7 (c), which has different shape from the charts on Figure 6 but shares similar characteristics. If we compare the projections of both clusterings, we can see that the relative positions between the clusters are similar. However, a clear difference appears in the general structure and, in particular, in the borders between the ■ cluster and the ■ cluster, where the separation becomes evident. We can see that the t-SNE projections of the clustering, $\mathcal{P}^g$, present many small subgroups within the bigger conglomerates, while the MDS projections tend to create very compact structures with dense yet well-structured borders. This picture clearly changes for the second clustering, $\mathcal{P}^c$, as we see in Figures 7 (d) and 7 (f). In particular, Figure 7 (d) strongly supports the existence of two main clusters: 1) the southern zone under the Pyrenees and 2) the northern area in conjunction with the Pyrenees. In this case, we can see more defined clusters and smaller borders between them. The difference is even clearer between the MDS projections (Figures 7 (e) and 7 (f)) where the two big groups are pushed to the sides with the clustering-based embedding f^c. The green clusters are now closer than before in relation to the rest of them. This means that the embedding f^c is able to either bring closer or separate the clusters and then, it can provide additional information about the underlying patterns of the sequence of images. On the other hand, the three clusters on the right of Figure 7 (f) have been slightly separated from each other. We can see that ■ and ■ clusters keep a similar connection, but the ■ cluster has been moved slightly away.



(a) Land according to $\mathcal{P}^g$ (b) Land according to $\mathcal{P}^c$

(c) t-SNE of $\mathcal{P}^g$ (d) t-SNE with $\mathcal{P}^c$

(e) MDS of $\mathcal{P}^g$ (f) MDS of $\mathcal{P}^c$

Fig. 7. Comparison between the geographic and embedded representation of $\mathcal{P}^g$ and $\mathcal{P}^c$.

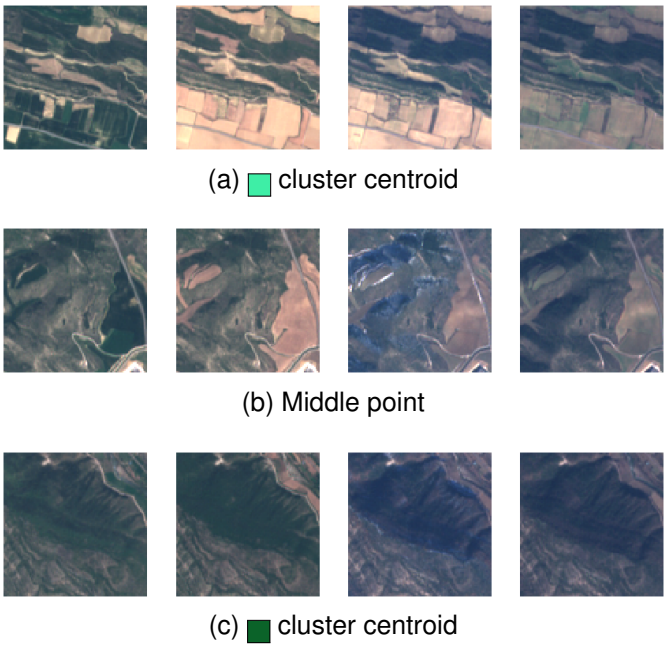


Fig. 8. Centroids and interpolation of the and clusters from $\mathcal{P}^c$.

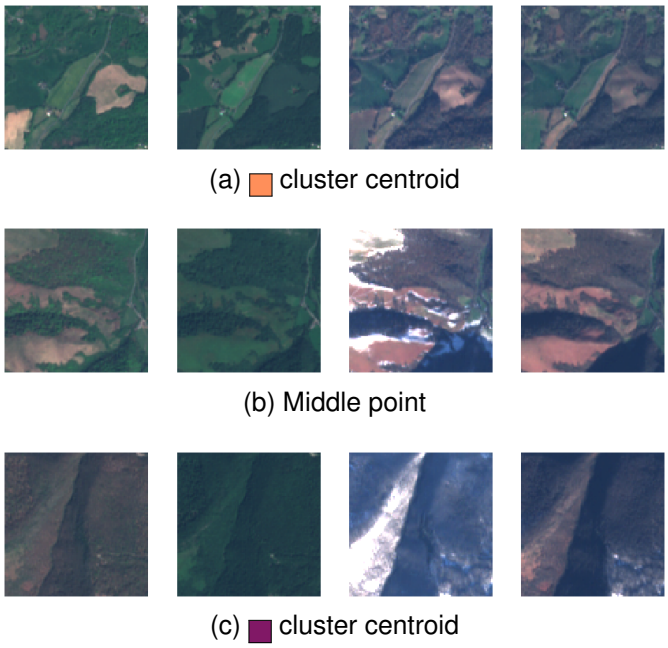


Fig. 10. Centroids and interpolation of the and clusters from $\mathcal{P}^c$.

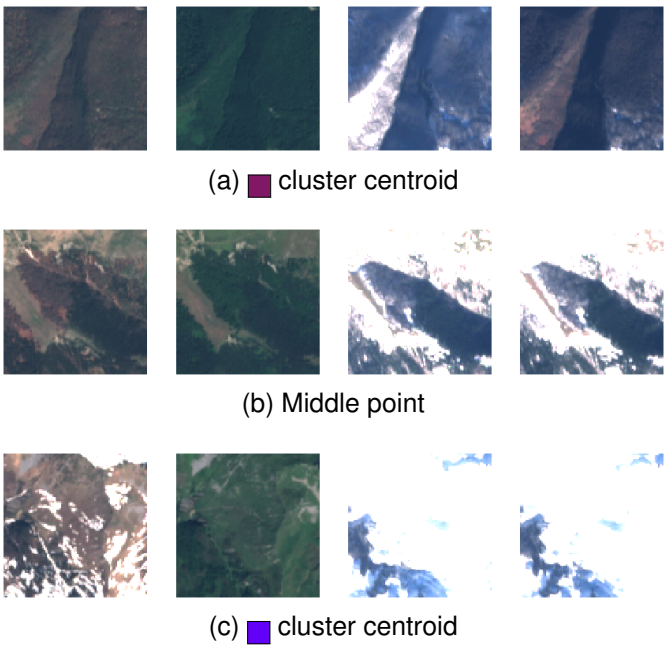


Fig. 9. Centroids and interpolation of the and clusters from $\mathcal{P}^c$.

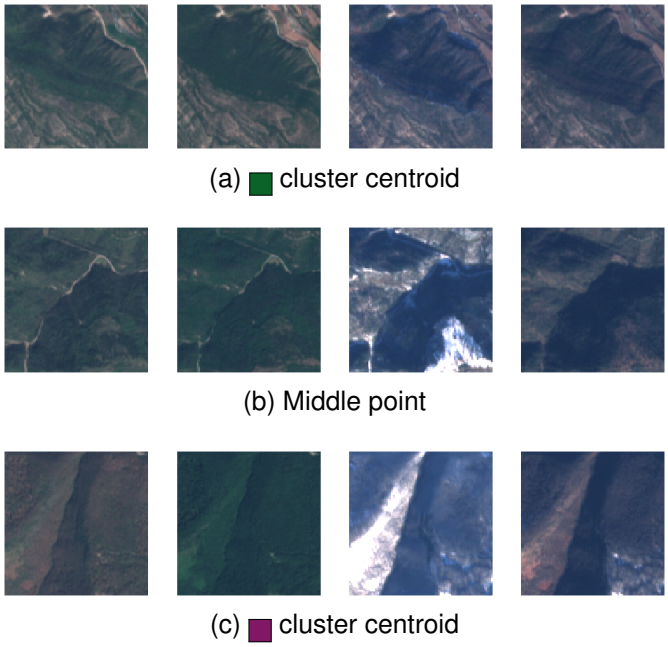


Fig. 11. Centroids and interpolation of the and clusters from $\mathcal{P}^c$.

C. Interpolation

We use the clustering, $\mathcal{P}^c$, given by the clustering-based embedding, f^c, to explore the spatio-temporal semantics that the clustering has captured. In order to do that, we show the closest ST to each centroid and the closest ST to some intermediate points between centroids in Figures 8, 9, 10, 11 and 12. Due to space limitations and also to facilitate the interpretation of the figures, we only show the first 4 elements of the STs that correspond to the four season of years 2017-2018. From left to right, the charts correspond to the spring, summer, autumn and winter.

The cluster centroids and the middle ST (interpolation with $w = 0.5$) between neighboring clusters are presented in Figures 8, 9, 10 and 11. The centroids (charts (a) and (c)) and the middle points (charts(b)) are shown together to better appreciate the changes in semantics. First, we can observe that each cluster centroid expresses a clear and well-defined semantic. Roughly speaking, the cluster is associated with crops and some pasture. The cluster includes different kinds of hills and pastures. The cluster also contains crops and pasture but of another type belonging to the north of the Pyrenees. The and clusters group together the mountains of the Pyrenees, where the cluster contains the highest

mountains with the most snow. We can see that, in these four figures, the interpolation with $w = 0.5$ always represents a semantic halfway between the two centroids. For instance, in Figure 8(b) we can see crops with more pasture than in Figure 8(a) and some hills that appear in Figure 8(c). In Figure 9(b) we observe more mountain peaks than in Figure 9(a) and an intermediate amount of snow in comparison with the centroids. Figure 10(b) shows crops between hills, which appear in in10(c). Finally, Figure 11(b) represents a landscape halfway between hills and mountains.

The last experiment is shown in Figure 12, where we present an interpolation with three intermediate steps, corresponding to $w \in \{0.25, 0.5, 0.75\}$. In this case, we choose the most distant clusters. The figure shows a smooth transition from the cluster, with crops along the year, to the cluster that contains high mountains with snow in some seasons of the year. We can see that the land contains more pasture and mountainous terrain as w increases, while the crops disappear. It is also worth mentioning that the amount of snow increases at each step.

The results of the clustering of STs go beyond the superficial details. For example, we have seen that the and clusters contain crop-related landscapes. Although these clusters apparently have similar semantics, they have been separated in the projection of the embedding. The difference between the crops on both sides of the Pyrenees have been captured by the MTS of embedded vectors and the subsequent clustering.

VII. Conclusions

In this paper, we have investigated a fully unsupervised methodology to conduct a semantic clustering over a region of interest from a sequence of satellite images. The sequence of images is encoded as a set of MTS by means of a semantically meaningful embedding, which is built in three steps: 1) training the embedding with triplets generated according to the geographic neighborhood, 2) clustering of the MTS and 3) embedding refining with triplets generated according to the clustering neighborhood.

The experiments are designed to explore the clustering from different perspectives in an unsupervised manner. Overall, the main conclusions of the paper are the following:

- The clustering of MTS based on embedded vectors exhibits robust and stable patterns of behavior in all the experiments carried out.
- The semantic clustering can contribute a wealth of knowledge about the region of interest, from coarse-grained semantic information to fine-grained details, as the number of clusters increases.
- There exist a very close connection between the geographic and the embedded representation of the clustering.
- The clustering can be refined and enhanced by means of a second phase of training based on the clustering neighborhood.
- The clustering of MTS automatically captures precise spatio-temporal semantic information.

We have seen that the geographic representation of the clustering exhibits a clear structure in which large areas are grouped in a very compact way. Nevertheless, as the number of clusters increases, a hierarchical partition naturally emerges, revealing an increasing number of semantic details, that could be studied more in-depth depending on the specific application. Both, in the geographic and the embedded representations, the clusters are arranged in a very similar pattern in terms of relative positions among them. Spatially adjacent clusters in the geographic representation tend to be adjacent in the embedding, and vice versa. Nevertheless, both spaces can provide complementary information about the shape of the clusters and their relationships. We have seen that the clustering-based embedding is able to sharpen the semantic information obtained from the sequence of satellite images. This embedding captures refined information about the underlying properties of the land for a given number of clusters. The clustering-based embedding highlights the elements' belonging to each cluster, i.e., it tries to separate the borders and brings the points to the center of the cluster, which would be desirable for further classifications tasks. The visual inspection of the centroids and the corresponding interpolations show the ability of the

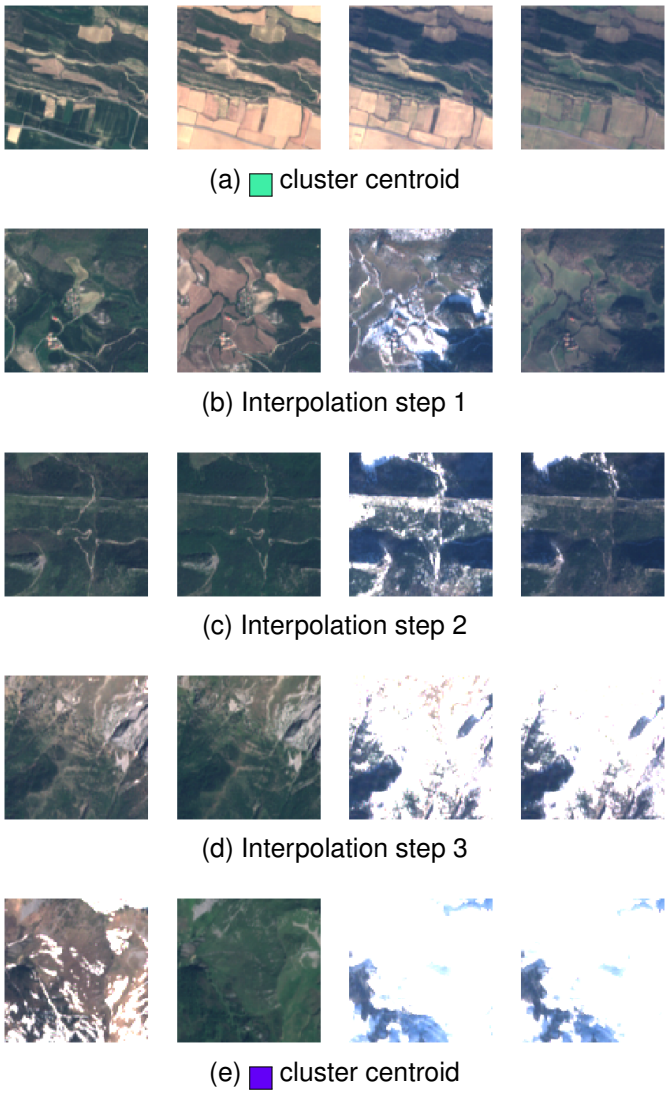


Fig. 12. Interpolation between and clusters with three intermediate points from $\mathcal{P}^c$.

clustering to capture the different semantics of a region and their evolution. Not only each cluster is able to represent a specific and well differentiated spatio-temporal semantic, but also the interpolation between clusters is clearly meaningful. The results indicate that the semantic properties investigated in previous works with isolated images and manually selected semantics are also expressed when a clustering of MTS is conducted.

We argue that the development and understanding of general unsupervised methods is crucial in the field of satellite images, where the labeled data is expensive to obtain. To illustrate our proposal, we have selected the region of Navarre in northern Spain, but any other region of interest could have been studied. The methods proposed in the current paper can later be incorporated into the pipeline of a larger system, where they can be combined with labeled data or expert knowledge if possible. Nonetheless,a fully unsupervised semantic analysis is a crucial tool to study a region of interest, as it allows to obtain from general patterns to specific details as the number of clusters increases. Thus, these kinds of methods can assist on a wide variety of issues such as studying the climate change, designing land policies or measuring human footprint.

Finally, it is important to note that the use of STs is an essential element to conduct further analysis related with the changing semantic of a region. The specific use of STs incorporates a new dimension in the semantic clustering that provides richer information and opens a wide variety of possibilities. For instance, the development of bi-clustering algorithms to be run over the set of MTS could find similar sub-sets both in space and time. Our proposal constitutes the fundamental basis to carry out these further developments, which would be highly suitable to detect change points and seasonality.

APPENDIX

A. The quality of partitional clustering from the Tile2Vec perspective

Once, the Tile2Vec embedding is learned from the tiles of a sequence of satellite images, we can represent the image as a grid of sequences tiles given by $\boldsymbol{Z} = \{\boldsymbol{z}_1, ..., \boldsymbol{z}_m\}$, where $\boldsymbol{z}_i = (z_i^1, ..., z_i^T)$ is a multidimensional time series of length T with $z_i^t \in \mathbb{R}^d$ for $i = 1, ..., m$ and $t = 1, ..., T$. A partitional clustering of $\boldsymbol{Z}$ is given by a set of subsets $\mathcal{P} = \{P_1, ..., P_K\}$ that is a partition of $\boldsymbol{Z}$, with $P_k \subset \boldsymbol{Z}$ and $P_k \neq \emptyset$. In particular, in this work, we propose the use of K-means algorithm to learn the partition $\mathcal{P}$. In this section, we motivate the use of this approach by relating the partitions learned using a K-means and an adaptation of the Tile2Vec error for partitional clustering.

The K-means algorithm is an iterative algorithm for partitional clustering heuristic that produces a sequence of clusterings until convergence. The algorithm deals with the minimization of the K-means error:

$$E(\mathcal{P}) = \sum_k \sum_{\boldsymbol{z} \in P_k} d(\boldsymbol{z}, \boldsymbol{c}_k)^2, \tag{3}$$

where $\boldsymbol{c}_k = \frac{1}{|P_k|} \sum_{\boldsymbol{z} \in P_k} \boldsymbol{z}$ is the centroid and $d(\boldsymbol{z}, \boldsymbol{z}') = \sum_t ||z_t - z'_t||_2$ is the Euclidean distance between the time series $\boldsymbol{z}$ and $\boldsymbol{z}'$. The minimization of the K-means error is NP-hard. It can be shown that, K-means algorithm reduces the K-means error every iteration until it converges to a stationary point.

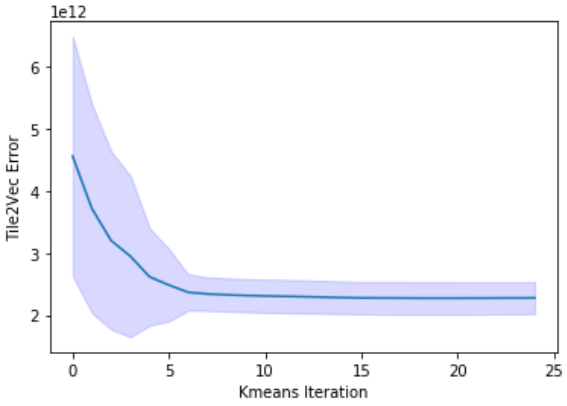


Fig. 13. The evolution of the average PT2V with the iterations of the K-means algorithm over 25 runs. The shadow region correspond to the standard deviation of the PT2V for each iteration.

We define the partitional Tile2Vec error (PT2V) for a clustering $\mathcal{P}$ as follows:

$$E(\mathcal{P}) = \sum_k \sum_{\boldsymbol{z}_a, \boldsymbol{z}_b \in P_k} \sum_{\boldsymbol{z}_c \in \boldsymbol{Z} \setminus P_k} [||\boldsymbol{z}_b - \boldsymbol{z}_a||_2 - ||\boldsymbol{z}_c - \boldsymbol{z}_a||_2 + m]_+ \tag{4}$$

where m is the margin of the standard Tile2Vec error function. The PT2V corresponds to the Tile2Vec error for the neighborhood defined by the partition $\mathcal{P}$, i.e., $\boldsymbol{z}$ and $\boldsymbol{z}'$ are neighbors when they belong to the same partition $P \in \mathcal{P}$, and otherwise they are considered distant. In other words, this error function defines the neighborhood in terms of the partition, and the distance of two sequences of tiles in the original image is irrelevant.

Figure 13 shows the evolution of the average PT2V for the clustering of $\boldsymbol{Z}$ obtained after each iteration, for 25 runs of the algorithm. On average, PT2V decreases monotonically with the iterations. This evidence shows a strong relation between the K-means error and Tile2Vec error, and motivates the use of the K-means algorithm to construct an appropriate partition $\mathcal{P}$ from the Tile2Vec error perspective.

REFERENCES

[1] C. Pelletier, S. Valero, J. Inglada, N. Champion, C. Marais-Sicre, and G. Dedieu, "Effect of training class label noise on classification performances for land cover mapping with satellite image time series," *Remote Sensing*, vol. 9, no. 2, 2017.

[2] O. Csillik, M. Belgiu, G. P. Asner, and M. Kelly, "Object-based time-constrained dynamic time warping classification of crops using Sentinel-2," *Remote Sensing*, vol. 11, no. 10, 2019.

[3] A. Sharifi, "Yield prediction with machine learning algorithms and satellite images." *Journal of the science of food and agriculture*, vol. 101, pp. 891–896, 2020.

[4] R. Negri, S. Sant'Anna, and L. Dutra, "Semi-supervised remote sensing image classification methods assessment," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 07 2011, pp. 2939–2942.

[5] S. Roy, E. Sangineto, N. Sebe, and B. Demir, "Semantic-fusion gans for semi-supervised satellite image classification," in *IEEE International Conference on Image Processing (ICIP)*, 2018, pp. 684–688.

[6] N. Jean, S. Wang, A. Samar, G. Azzari, D. Lobell, and S. Ermon, "Tile2vec: Unsupervised Representation Learning for Spatially Distributed Data," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 3967–3974, Jul. 2019.

[7] E. Kalinicheva, J. Sublime, and M. Trocan, "Unsupervised Satellite Image Time Series Clustering Using Object-Based Approaches and 3D Convolutional Autoencoder," *Remote Sensing*, vol. 12, no. 11, 2020.

[8] J. Wang, C. H. Q. Ding, S. Chen, C. He, and B. Luo, "Semi-supervised remote sensing image semantic segmentation via consistency regularization and average update of pseudo-label," *Remote Sensing*, vol. 12, no. 21, 2020. [Online]. Available: https://www.mdpi.com/2072-4292/12/21/3603

[9] T. Guyet and N. Hervé, "Long term analysis of time series of satellite images," *Pattern Recognition Letters*, vol. 70, pp. 17–23, 2016.

[10] Z. Zhang, P. Tang, L. Huo, and Z. Zhou, "Modis ndvi time series clustering under dynamic time warping," *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 12, 2014.

[11] Z. Zhang, P. Tang, W. Zhang, and L. Tang, "Satellite image time series clustering via time adaptive optimal transport," *Remote Sensing*, vol. 13, no. 19, 2021.

[12] S. Li, W. Song, L. Fang, Y. Chen, P. Ghamisi, and J. Benediktsson, "Deep learning for hyperspectral image classification: An overview," *IEEE Transactions on Geoscience and Remote Sensing*, vol. PP, pp. 1–20, 04 2019.

[13] W. R. Moskolaï, W. Abdou, A. Dipanda, and Kolyang, "Application of deep learning architectures for satellite image time series prediction: A review," *Remote Sensing*, vol. 13, no. 23, 2021.

[14] Z. I. Petrou and Y. Tian, "Prediction of sea ice motion with convolutional long short-term memory networks," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6865–6876, 2019.

[15] I. Prapas, S. Kondylatos, I. Papoutsis, G. Camps-Valls, M. Ronco, M. Fernández-Torres, M. P. Guillem, and N. Carvalhais, "Deep learning methods for daily wildfire danger forecasting," *CoRR*, vol. abs/2111.02736, 2021. [Online]. Available: https://arxiv.org/abs/2111.02736

[16] J. Jagannathan and C. Divya, "Deep learning for the prediction and classification of land use and land cover changes using deep convolutional neural network," *Ecological Informatics*, vol. 65, p. 101412, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S157495412100203X

[17] S. Wang, S. Di Tommaso, J. Faulkner, T. Friedel, A. Kennepohl, R. Strey, and D. B. Lobell, "Mapping crop types in southeast india with smartphone crowdsourcing and deep learning," *Remote Sensing*, vol. 12, no. 18, 2020.

[18] C. Pelletier, G. I. Webb, and F. Petitjean, "Temporal convolutional neural network for the classification of satellite image time series," *Remote. Sens.*, vol. 11, p. 523, 2019.

[19] M. Piles, J. Muñoz-Marí, A. Guerrero-Curieses, G. Camps-Valls, and J. L. Rojo-Álvarez, "Autocorrelation metrics to estimate soil moisture persistence from satellite time series: Application to semiarid regions," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–17, 2022.

[20] S. Ji, C. Zhang, A. Xu, Y. Shi, and Y. Duan, "3d convolutional neural networks for crop classification with multi-temporal remote sensing images," *Remote Sensing*, vol. 10, no. 1, p. 75, 2018.

[21] X. He, Y. Zhou, J. Zhao, D. Zhang, R. Yao, and Y. Xue, "Swin transformer embedding unet for remote sensing image semantic segmentation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 60, pp. 1–15, 2022.

[22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations (ICLR), Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2013.

[23] U. Pérez-Goya, M. Montesino-SanMartin, A. F. Militino, and M. D. Ugarte, "rsat: Dealing with multiplatform satellite images from Landsat, MODIS, and Sentinel. R package version 0.1.16." https://github.com/ropensci/rsat, 2021.

[24] C. Storie and C. J. Henry, "Deep learning neural networks for land use land cover mapping," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 07 2018, pp. 3445–3448.

[25] Z. Zhang, P. Tang, W. Zhang, and L. Tang, "Satellite Image Time Series Clustering via Time Adaptive Optimal Transport," *Remote Sensing*, vol. 13, no. 19, 2021.

[26] T. Lampert, B. Lafabregue, and P. Gançarski, "Constrained distance based k-means clustering for satellite image time-series," in *IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, 2019, pp. 2419–2422.

[27] 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," *International Journal of Applied Earth Observation and Geoinformation*, vol. 74, pp. 103–119, 2019.

[28] P. Jenkins, A. Farag, S. Wang, and Z. Li, "Unsupervised representation learning of spatial data via multimodal embedding," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1993–2002.

[29] J. Kang, R. Fernandez-Beltran, P. Duan, S. Liu, and A. J. Plaza, "Deep unsupervised embedding for remotely sensed images based on spatially augmented momentum contrast," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 3, pp. 2598–2610, 2021.

[30] J. Bjorck, B. H. Rappazzo, Q. Shi, C. Brown-Lima, J. Dean, A. Fuller, and C. Gomes, "Accelerating ecological sciences from above: Spatial contrastive learning for remote sensing," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 17, pp. 14 711–14 720, May 2021.

[31] G. Taskin and G. Camps-Valls, "Graph embedding via high dimensional model representation for hyperspectral images," *IEEE Transactions on Geoscience and Remote Sensing*, pp. 1–1, 2021.

[32] G. Cheng, C. Yang, X. Yao, L. Guo, and J. Han, "When deep learning meets metric learning: Remote sensing image scene classification via learning discriminative cnns," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 56, pp. 2811–2821, 2018.

[33] S. Woźniak and P. Szymański, *Hex2vec: Context-Aware Embedding H3 Hexagons with OpenStreetMap Tags*. Association for Computing Machinery, 2021, p. 61–71.

[34] H. Jung, Y. Oh, S. Jeong, C. Lee, and T. Jeon, "Contrastive self-supervised learning with smoothed representation for remote sensing," *IEEE Geoscience and Remote Sensing Letters*, vol. 19, pp. 1–5, 2022.

[35] T. Chen, S. Kornblith, M. Norouzi, and G. Hinton, "A simple framework for contrastive learning of visual representations," in *Proceedings of the 37th International Conference on Machine Learning*, 2020, pp. 1597–1607.

[36] M. C. Hansen, P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland, A. Kommareddy, A. Egorov, L. Chini, C. O. Justice, and J. R. G. Townshend, "High-resolution global maps of 21st-century forest cover change," *Science*, vol. 342, no. 6160, pp. 850–853, 2013.

[37] G. Rousset, M. Despinoy, K. Schindler, and M. Mangeas, "Assessment of deep learning techniques for land use land cover classification in southern new caledonia," *Remote Sensing*, vol. 13, no. 12, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/12/2257

[38] M. Debella-Gilo and A. K. Gjertsen, "Mapping seasonal agricultural land use types using deep learning on Sentinel-2 image time series," *Remote Sensing*, vol. 13, no. 2, 2021. [Online]. Available: https://www.mdpi.com/2072-4292/13/2/289

[39] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A neural probabilistic language model," *J. Mach. Learn. Res.*, vol. 3, p. 1137–1155, mar 2003.

[40] W. R. Tobler, "A computer movie simulating urban growth in the detroit region," *Economic Geography*, vol. 46, pp. 234–240, 1970.

[41] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778.

[42] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*, 2013.

[43] R. Santana, *Semantic Composition of Word-Embeddings with Genetic Programming*. Cham: Springer International Publishing, 2021, pp. 409–423. [Online]. Available: https://doi.org/10.1007/978-3-030-58930-1_27

[44] S. Lloyd, "Least squares quantization in PCM," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[45] X. Wu, V. Kumar, J. Ross Quinlan, J. Ghosh, Q. Yang, H. Motoda, G. J. McLachlan, A. Ng, B. Liu, P. S. Yu *et al.*, "Top 10 algorithms in data mining," *Knowledge and information systems*, vol. 14, no. 1, pp. 1–37, 2008.

[46] L. van der Maaten and G. Hinton, "Viualizing data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579–2605, 11 2008.

[47] J. B. Kruskal, "Nonmetric multidimensional scaling: a numerical method," *Psychometrika*, vol. 29, no. 2, pp. 115–129, 1964.