

Article

Representation learning of multispectral Earth Observation time series and evaluation for crop type classification

Andrea González-Ramírez ^{1,*}, Clement Atzberger ², Deni Torres-Roman ¹ and Josué López ²

¹ Center for Research and Advanced Studies of the National Polytechnic Institute, Telecommunications Group, Av del Bosque 1145, Zapopan 45017, Mexico

² Mantle Labs, Grünentorgasse 19, Vienna 1090, Austria

* Correspondence: andrea.gonzalez@cinvestav.mx

Abstract: Remote sensing (RS) spectral time series provide a substantial source of information for the regular and cost-efficient monitoring of the Earth surface. Important monitoring tasks include land use and land cover classification, change detection, forest monitoring, crop type identification, among others. To develop accurate solutions for RS-based applications, often supervised shallow/deep learning algorithms are used. However, such approaches usually require fixed-length inputs and large labeled datasets. Unfortunately, RS images acquired by optical sensors are frequently degraded by aerosol contamination, clouds and cloud shadows, producing missing observations and irregular observation patterns. To address these issues, efforts have been made to implement frameworks that generate meaningful representations from the irregularly sampled data streams and alleviate the deficiencies of the data sources and supervised algorithms. Here, we propose a conceptually and computationally simple representation learning (RL) approach based on autoencoders (AEs) to generate discriminative features for crop type classification. The proposed methodology ensembles a set of single layer AEs with very limited number of neurons, each one trained with mono-temporal spectral features of a small set of samples belonging to a class, resulting in a model capable of processing very large areas in low computational time. Importantly, the developed approach remains flexible with respect to the availability of clear temporal observations. The signal derived from the ensemble of AE is the reconstruction difference vector between input samples and their corresponding estimations, which are averaged over all cloud/shadows-free temporal observations of a pixel location. This averaged reconstruction difference vector is the base for the representations and the subsequent classification. Experimental results show that the proposed extremely light-weight architecture indeed generates separable features for competitive performances in crop type classification, as distance metrics scores achieved with the derived representations significantly outperform those obtained with the initial data. Conventional classification models were trained and tested with representations generated from a widely used Sentinel-2 multispectral multitemporal dataset, BreizhCrops. Our method achieves 76.72% overall accuracy which is ~ 6% higher than using original Sentinel-2 data within conventional classifiers and even ~ 4% better than complex deep models as OmnisCNN. Compared to extremely complex and time-consuming models such as transformers and LSTMs, only a 3% reduction in overall accuracy was noted. Our method uses only 6.8k parameters, i.e., ~ 400x less than OmnisCNN and ~ 27x less than Transformers. The results prove that our method is competitive in classification performance compared with state-of-the-art methods while requiring much lower computational load.

Citation: Lastname, F.; Lastname, F.; Lastname, F. Title. *Journal Not Specified* 2023, 1, 0. <https://doi.org/>

Received:

Revised:

Accepted:

Published:

Copyright: © 2024 by the authors. Submitted to *Journal Not Specified* for possible open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: crop types; multispectral; multitemporal; autoencoder; representation learning

1. Introduction

Spectral observations of the Earth's surface using remote sensors have been used since long time for crop type mapping, given the quantity and availability of spectral-temporal images. Multi-spectral time series from sensors such as Landsat or Sentinel-2

have provided very cost-effective technical support to achieve the reliable identification and monitoring of large cropping areas [1–6]. While a wide number of data sources and supervised classification algorithms have been used for crop mapping [4,7–15], limited efforts have been made in feature learning as well as the use of un- and self-supervised learning algorithms to alleviate missing data produced by clouds. Notable overviews and examples are provided in [16–20].

The use of temporal series of multi-spectral observations for crop type classification is advantageous as the spectral differences in the crop growth, composition and structure over time are exploited [1,6,17,21]. Each crop type has a distinct seasonal spectral behavior depending on local weather and growth conditions [3,6,12]. Therefore, many researchers center their works on making use of multi-temporal information instead of using single acquisitions [1,10,22,23].

The most common methods for crop type classification are based on supervised learning algorithms [4,12,14,15,21,24–32]. The aim of these algorithms is to train a discriminative model using labeled data. However, it is often complicated to find tagged datasets for the region of interest, since it requires human accurate intervention. Examples of supervised machine learning (ML) models include decision trees (DT) [33], extreme gradient boosting (XGBoost) [34], random forest (RF) [35], support vector machine (SVM) [36], and artificial neural networks (ANN) [37]. The mentioned algorithms provide usually similar classification performance, but often require extensive preprocessing steps such as compositing and gap-filling when incomplete (e.g., cloud-corrupted) time series are analyzed.

To mitigate the reliance on large labeled datasets, unsupervised learning aims to first derive (latent) representations from the abundant unlabeled spectral data. Representation learning (RL) is a broad subfield in machine learning, which is a set of techniques focused on automatically learning and identifying meaningful features from the input data. The derived representations encode the internal structure of the data, so that any subsequent classification needs fewer labels to be trained. In extreme cases this leads to approaches such as few-shot learning or even one-shot learning. To derive representations that efficiently encode the original data, a large number of algorithms have been developed over the past years, as for example summarized in the work of Balestrieri et al. [38].

To cope with missing data in temporal observations, within the field of representation learning, different approaches have been developed as we will outline in subsection 1.1.

Autoencoders (AE) have as objective to compress data into a lower dimensional space, known as code, and then to reconstruct the input [39]. The code is regarded to be a set of features, also called representations, which condense the necessary information to recover the original data [40]. If spectral observations from a given location/pixel are tagged with the corresponding information regarding the time of observation (e.g., day of year), an autoencoder can in principle also learn to encode inputs along the time axis.

Typically, this feature of AEs is used for change detection, where the sought events are seen as anomalies in the reconstruction difference [41–43]. In technical terms, this can be framed as if the event-specific observations depart from the “normal” object-specific manifold within the embedding space. The use of ensembles of AE - each trained on different object-classes – where the resulting vectors of temporal reconstruction differences are subsequently used for classification purposes, has not been widely studied.

In this work, we propose to train a light-weight deep learning model with individual time-tagged spectral signatures, while bypassing gap-filling and compositing methods. In our framework, we use an ensemble of AEs to generate new informative and discriminative features. The features are evaluated in this work with respect to a crop type classification. Here, we arbitrarily chose one simple AE per class, but other choices would also be possible. We calculate the AEs reconstruction difference vectors between input and output and concatenate them to form a vector of representations. We evaluate the performance of the derived representations by comparing classification performance using as input data Sentinel-2 time series and the representations generated by our method on conventional classifiers, RF, SVM, XGBoost and a simple fully connected network (FCN). In addition, we

compare the outcomes against a number of more complex benchmark approaches using
91
the same dataset.
92

1.1. Related work

93

Russwurm et al. [12] presented a satellite image time series dataset for crop type
94 mapping named BreizhCrops. They generated top- (TOA) and bottom-of-atmosphere
95 (BOA) time series from Sentinel-2 and used the dataset to benchmark seven classifiers
96 for crop type mapping. A particularity of this dataset is the extremely limited number of
97 samples for certain classes, as two minority crop types (sunflower and nuts) have much less
98 samples than the other classes. This challenges model's capacity to effectively generalize
99 to underrepresented classes. The difficulty to correctly classify the minority classes was
100 even noted by the authors for high complexity benchmark models such as transformer and
101 long-short term memory (LSTM) approaches.
102

Paris et al. [11] proposed an approach based on a LSTM model. They addressed the
103 problem of cloud-corrupted multitemporal data by constructing a large training dataset
104 from three full Sentinel-2 tiles, with orbital overlap area, to create monthly composite
105 images. However, this approach depends on numerous cloud-clear temporal observations
106 to generate trustworthy composites. Moreover, inaccurate cloud masks induce incorrect
107 composite values, which compromises classification performance. The practical usage of
108 this approach is mainly limited due to the demand for large computational resources that
109 high complexity models require.
110

He et al. [18] proposed a crop type classification method, trying to improve mod-
111 els performance by merging spectral, textural and environmental features. One major
112 downside of this approach is that while feature learning/selection methods attempt to
113 reduce data redundancy, combining a large number of features easily induces redundancy,
114 affecting classification performance. Furthermore, collecting and processing these addi-
115 tional features is time-consuming and computationally expensive, potentially limiting the
116 method's scalability in larger or more diverse locations.
117

Lisaius et al. [44] proposed a novel representation learning approach for remote
118 sensing data based on a twins network. They derive representations from a spectral-
119 temporal Barlow Twin (STBT) and afterwards assess the quality of the representations
120 within a supervised crop type classification. This method uses sparse temporal sampling
121 as the only augmentation strategy addressing cloud-corruption issues. However, the lack
122 of additional augmentation types restricts the model's capacity to manage other types of
123 data corruption. As other approaches, this method assumes that cloudy observations are
124 totally removed from the data, which is not optimum in real-world circumstances with
125 poor quality cloud masks.
126

Kalinicheva et al. [43] proposed a particularly interesting approach with AEs. Recon-
127 struction losses of joint AEs are used to detect non-trivial changes between two co-registered
128 images in a satellite image time series. This method depends on patch-wise reconstruction
129 error, and hence the approach has difficulty capturing fine features for objects that are
130 only 1-2 pixels wide. Moreover, joint autoencoder models, particularly convolutional
131 autoencoders, need significantly large training time, which makes the method unsuitable
132 for real-time or large-scale applications.
133

Windrim et al. [45] proposed an approach using AEs for unsupervised feature-learning
134 with hyperspectral data. The method allows to evaluate the separability of the feature
135 spaces for clustering tasks. Hyperspectral data are naturally high-dimensional, and this
136 work recognizes that high-dimensional data present issues such as greater data variability
137 and computational complexity.
138

Table 1. Summary of relevant works related to crop types classification and representation learning.

References(year)[cites]	Satellite	Time range	Method	Number of classes	Feature selection
Kalinicheva, E., et al, (2019)[19] [43].	SPOT-5	2002 2008	AEs	Not specified	N/A
Windrim, E., et al, (2019)[42] [45].	AVIRIS and others	Not specified	AEs	Not specified	N/A
Paris, Claudia, et al (2020)[13] [11].	S2	09/2017 08/2018	LSTM	12	N/A
Russwurm, Marc, et al. (2020)[65]. [12]	S2	01/01/2017 31/12/2017	ANNs	9	N/A
Zhiwei Yi, et al. (2020)[56]. [17]	S2	23/04/2019 20/09/2019	RF	8	Spectro temporal
Shan He, et al. (2022)[3]. [18]	MODIS	01/01/2009 31/12/2009	KS	4	Spectral textual environmental
Leikun Yin, et al. (2020)[35]. [16]	S2	01/04/2018 31/10/2018	RF	3	Spectro temporal
Lisaius, et al. (2024)[-]. [44]	S2	01/01/2017 31/12/2018	STBT	8	Spectro temporal
Proposal in this work	S2	01/01/2017 31/12/2017	AEs	9	N/A

Other approaches in the state of the art address the problem of missing data with combination of optical and Synthetic Aperture Radar (SAR) data [5,24,46,47], fusion of multiple sensors [8,48,49], data interpolation [12,50] or simply using only a subset of partially cloud free observations as Zhiwei et al [17] and Shan et al [18].

In summary, major issues in the state of the art are: 1) use of highly complex models, 2) infeasibility to scale to large areas, 3) the reliance on interpolation methods, 4) dependency on reliable cloud masks, and 5) handling of huge amount of data. Most of the machine learning-based approaches need extremely deep models leading to high computational costs and processing time, and therefore limiting scalability to process large areas of interest. Approaches that rely on interpolation methods exploit the smooth changes between data points, but fail if data gaps become overly long. Many approaches also require fixed-length sequences, which restricts the model's flexibility in dealing with irregular inputs. Sensor fusion approaches, on the other hand, face the challenge of handling huge amounts of data, which leads to high computational load and increases in processing time.

1.2. Contributions

The main contributions of this work are the following:

1. To tackle cloud-corrupted time series analysis, the proposed framework processes individual time-tagged spectral signatures for feature extraction and thereby completely avoids the use of gap-filling and compositing methods.
2. The proposed methodology uses neural networks with a reduced number of neurons to keep the computational load low, thereby facilitating the processing of large geographic areas.
3. The proposed pixel-wise framework provides a robust solution with respect to the number of available cloud-free observations, while achieving competitive results even under high levels of cloudiness. By avoiding the use of spatial convolutions, the approach focuses on the information within the specific pixel location and thus can also be applied to regions with very small object sizes.

The remainder of this work is organized as follows. Section 2 presents the concept of RL and the respective mathematical definitions, as well as a brief description of AEs. In Section 3 the problem statement of this work and the mathematical formulation of the proposed framework are introduced. Section 4 describes quantitative and qualitative

experimental results. Sections 5 and 6 present the discussion and conclusions, respectively, of the results obtained in our experiments.

2. Materials and Methods

2.1. Representation Learning

Representation learning (RL), also called feature learning, is a subfield of machine learning that aims to automatically learn and identify meaningful features, or representations, from the input data. Representations are expected to be more informative for downstream tasks such as clustering, regression, or classification [51].

In crop type classification, RL provides various benefits, especially in agricultural applications, where managing high-dimensional and complex dataset is crucial [52]. RL models first learn hierarchical features, beginning with low-level details such as edges and textures and advancing to higher-level characteristics like specific crop patterns. This hierarchy is critical for differentiating between different crop varieties [24]. Furthermore, representations derived with RL from the abundant unlabeled data mitigate the reliance on large labeled datasets, which is usually costly and time-consuming to produce in agriculture.

RL has therefore the potential to improve crop classification by automating feature extraction, enhancing generalization, reducing the need for large labeled datasets, and adjusting to environmental variability. With its scalability, speed, and capacity to combine geographical and temporal data, it is an effective tool for developing robust, accurate, and scalable crop classification models.

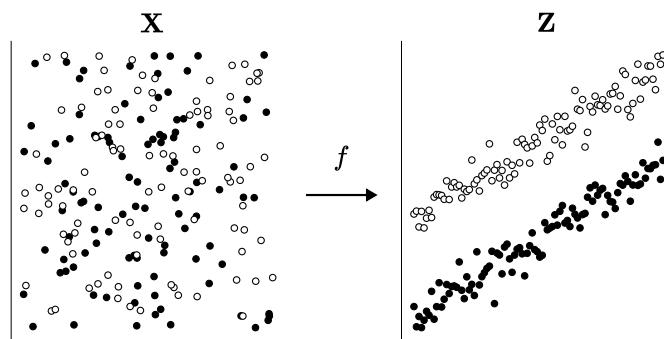


Figure 1. Illustration of representation learning (RL) as a function f , mapping vectors from a dimensional space to a representation space.

Mathematically, RL is defined as a function $f : \mathbf{X} \rightarrow \mathbf{Z}$, that transforms the input data $\mathbf{X} = \{\mathbf{x}_1, \dots, \mathbf{x}_S\}$, into features $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_S\}$, where each vector $\mathbf{x}_s \in \mathbb{R}^n$ and its image $\mathbf{z}_s \in \mathbb{R}^p$, and \mathbb{R}^p denotes the representation space. The objective function f leads the model to learn meaningful representations of the input data, preserving information, reducing redundancy and generally reducing dimensionality.

In recent years, many RL methods have been proposed from different perspectives and families [38], e.g., contrastive learning methods (InfoNCE [53–55]), deep metric learning (SimCLR [56,57], NNCLR [58], etc.), non-contrastive methods (VICReg [59], Barlow Twins [44,60], etc), among others. Such approaches are particularly useful in cases where the observed data is generated by a limited set of variables [61]. However, RL is not limited to these families of methods, and conventional neural network models, such as autoencoders (AEs) can also form a representation learning method.

These approaches can also be seen as belonging to the field of self-supervised learning (SSL). Indeed, self-supervised learning techniques enable models to be pre-trained on unlabeled data, reducing reliance on labeled datasets. Furthermore, data augmentation techniques (rotations, translations, etc.) applied in these type of models have proven to improve performance without increasing the amount of training data.

2.2. Autoencoders

Autoencoders are a specific type of ANN used for unsupervised learning (Figure 2) [62,63]. They have applications in various research fields, such as anomaly detection, data compression, and feature learning. Their aim is to encode the input into a compressed representation, and then reconstruct the input from this representation, so that the reconstruction is as similar as possible to the input [51,64].

Although AEs are not in principle designed for detection and classification tasks, several works have demonstrated their potential to ease these tasks by using AE-derived data for change detection and binary classification models [41–43]. Since AEs are trained to compress and afterwards reconstruct the input data, they basically learn to model samples belonging to a certain joint distribution, leading the model to learn class-specific properties.

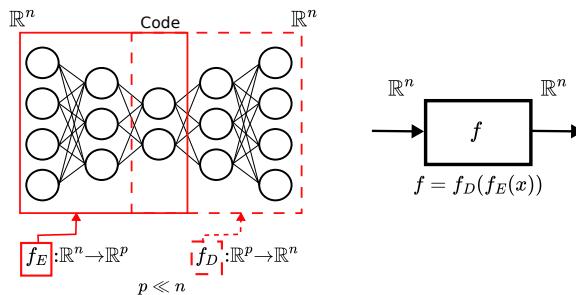


Figure 2. Example of an autoencoder architecture with mathematical definition as a function. In the present work, the reconstruction difference between input and output is used as representation, and not the code itself.

The aim of the AEs, formally defined in [65], is to learn the functions $f_E : \mathbb{R}^n \rightarrow \mathbb{R}^p$ and $f_D : \mathbb{R}^p \rightarrow \mathbb{R}^n$, where f_E denotes the encoder function, f_D is the decoder function, n is the dimension of the input and output spaces, and p denotes the dimension of the code space. Generally $p \ll n$, leading to learn compressed features of the data.

3. Proposed method

Consider an annual multi-spectral time series dataset acquired by an optical sensor, i.e., each sample has been acquired at different times. From the entire set of observations, only a subset will usually be useful as weather conditions such as clouds, cirrus, cloud shadows, snow, among others, occasionally obstruct the land surface. Missing data produced by these conditions commonly leads to poor performance on particular tasks, such as land use / land cover classification or change detection. Therefore, it is of utmost importance to extract and use only the land related information, either by filtering the data, or generating new features (often in the form of composites).

3.1. Problem statement

Let $\mathcal{X} \in \mathbb{R}^{P \times B \times T}$ be a multispectral time series dataset represented as a third-order array, where P represents the number of geographic points on the earth surface, B is the number of spectral bands, and T denotes the number of temporal observations, and each geographic point is denoted as a vector $\mathbf{x} \in \mathbb{R}^{B \cdot T}$ and $\mathbf{x} \in \mathcal{X}$. The aim is to transform each vector \mathbf{x} into a representation vector $\mathbf{z} \in \mathbb{R}^R$, where R is the number of new features named representations. The representation vector \mathbf{z} addresses label scarcity and missing data produced by clouds, and permits downstream tasks such as crop type classification (See Figure 3).

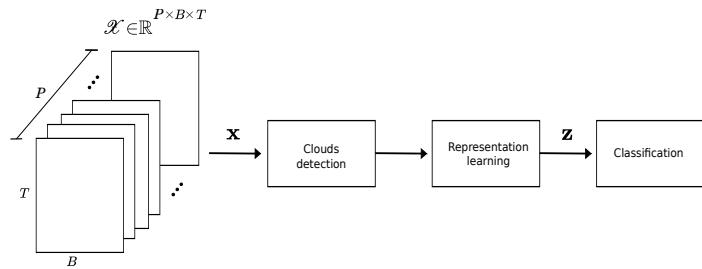


Figure 3. First level of the proposed workflow. Scene classification product provided by the European Space Agency (ESA) is used to mask out cloudy samples from a geographic point (pixel) shaped as a $T \times B$ array.

3.2. Methodology

The methodology of this work consists of four processes: data downloading and preprocessing, model training, inference (representations formation) and, as downstream task to evaluate the quality of the derived representations, classification. The proposed framework is shown in more detail in Figure 4.

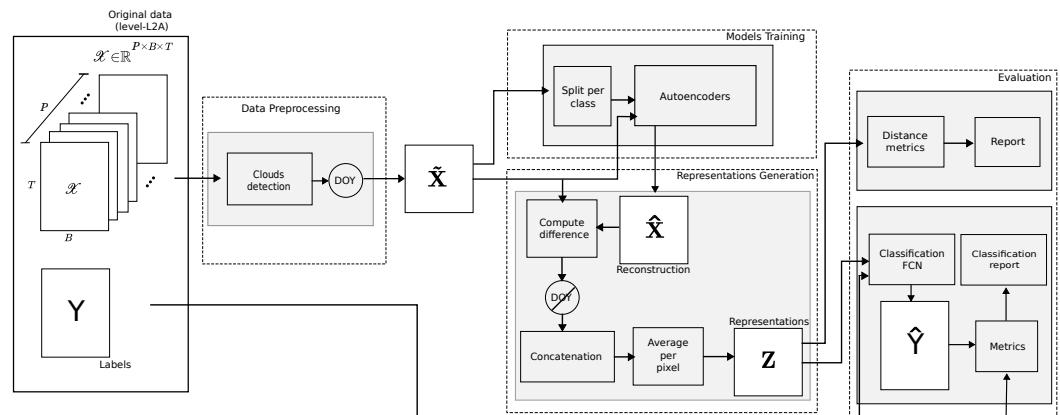


Figure 4. Proposed framework block diagram. The full methodology is composed by four main blocks: data preprocessing, model training, representation generation and evaluation.

3.2.1. Data downloading/preprocessing

Reference crop type labels were extracted from a public benchmark dataset named BreizhCrops [12] (field level). Google earth engine (GEE) was used to download full multitemporal multispectral data from a region of interest (ROI) (see Figure 5).

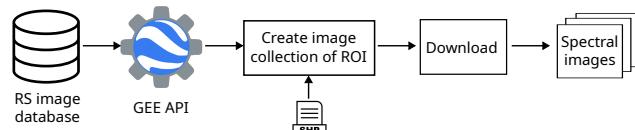


Figure 5. Dataset downloading process using the google earth engine (GEE) database.

Given the multispectral time series, a scene classification product (e.g., sen2cor) is used to get a cloud/non-cloud mask for each sample x , i.e., for each temporal observation at pixel level. Cloudy samples are excluded from the dataset. This creates a set of pixels with variable number of cloud free (temporal) observations. Only the cloud free observations of a pixel are the inputs to generate a representations vector. This makes our model flexible and independent on the number of clear observations.

To leverage the temporal information for the particular task of crop type classification, we add temporal embeddings to each sample, with the aim to extend the vector space to

one where similar spectral curves of different crop types at different growth stages are separable. We use the sine and cosine functions to model the annual periodic phenomenon presented by the cyclical character of the nature seasons and crops evolution between planting and harvest. As day 1 and 365 are in principle distant but with similar natural conditions, we scaled the acquisition day-of-year (DOY) to $(0, 1)$ range dividing by 365 and then place them on a real value scale by computing the sine and cosine. This makes each scaled DOY a unit vector decomposed into two orthogonal vectors, regarded as spring–fall axis (sine), and summer–winter axis (cosine) giving to our method the capacity to perform correctly in different earth latitudes [66].

Hence, each sample $\tilde{\mathbf{x}} \in \mathbb{R}^F$ has F features, i.e., B spectral bands plus two values denoting the sensing DOY, computed as follows

$$doy_{\sin} = \left(\sin\left(\frac{2\pi doy}{365}\right) + 1 \right) / 2 \quad (1)$$

and

$$doy_{\cos} = \left(\cos\left(\frac{2\pi doy}{365}\right) + 1 \right) / 2, \quad (2)$$

where doy denotes the DOY as a numeric value from 1 to 365, and doy_{\sin} and doy_{\cos} are in the range 0 to 1.

3.2.2. Model training

The principle of this work is to train a set of C independent AEs with vectors $\tilde{\mathbf{x}}_c$, which are individual time-tagged spectral signatures of cloud/shadows-free observations that belong to a particular class/cluster c for $c = 1, \dots, C$, resulting in C semi-supervised trained models able to reconstruct samples from the same class/cluster, approaching the reconstruction difference vector to zero, while using the ensemble of reconstruction difference vectors to derive the representations (see Figure 6).

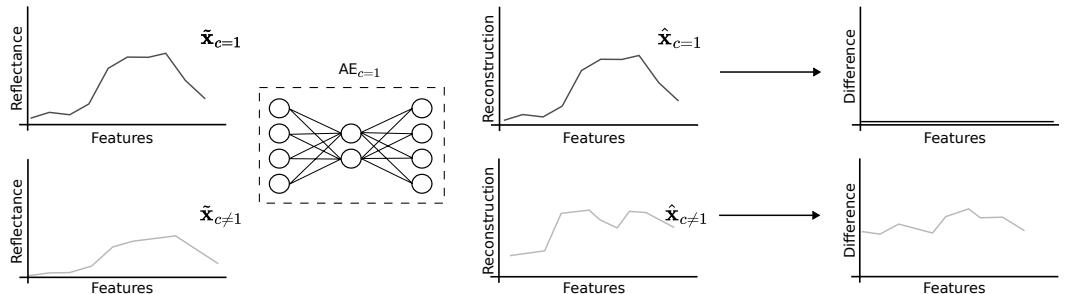


Figure 6. Example of the expected output for positive and negative samples. The "difference" from the ensemble of autoencoders (AEs) constitute the representations for the downstream task.

The training process of the AEs can be semi-supervised, given a labeled dataset, as in this work, or unsupervised, with no ground truth data (e.g., by training a set of random AEs not associated to specific crop types or classes). The scope of this work addresses the semi-supervised approach, with a crop type labeled dataset as pairs (x, y) , where y is an integer value which indicates the class that x belongs to. Samples are split in as many subdatasets as classes and each subdataset is used to train a different AE. This process is graphically represented in Figure 4 as model training and further illustrated in Figure 7.

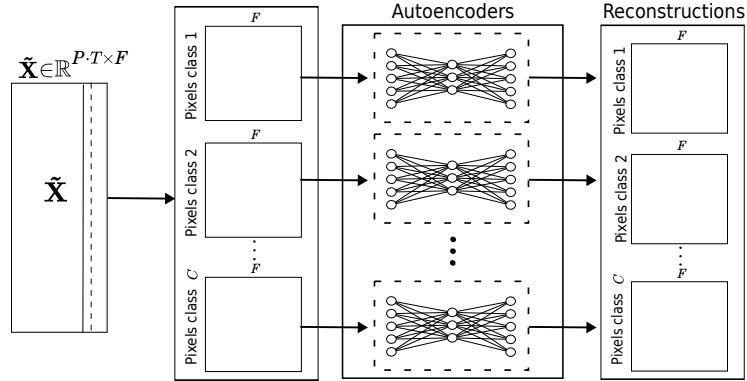


Figure 7. Autoencoders (AEs) training. Each autoencoder is trained with a set of individual spectral curves belonging to one of the crop types. The reconstructions from the C classes are used to calculate the difference vector across the ensemble, that is the final set of representations.

As mentioned in Section 2.2, an AE is formally defined as a composed function $f(x) = f_D(f_E(x))$. For our input vectors $\tilde{\mathbf{x}}_c$, each AE can be seen as a function

$$f(\tilde{\mathbf{x}}_c) = f_D(f_E(\tilde{\mathbf{x}}_c)) \quad (3)$$

where, $f_E(\tilde{\mathbf{x}}_c)$ denotes the encoder function, which maps the input vectors that belong to class c , from the input space \mathbb{R}^F to an embedding space \mathbb{R}^P , known as code, and, at the same time, is a composed function of the hidden layers f_{E_i}

$$f_E(\tilde{\mathbf{x}}_c) = f_{E_I}(f_{E_{I-1}}(\dots(f_{E_1}(\tilde{\mathbf{x}}_c)))) \quad (4)$$

where $f_{E_i}(\cdot)$, for $i = 1, \dots, I$ are regression functions

$$f_{E_i}(\mathbf{y}_{i-1}) = \alpha_i(\mathbf{W}_i \mathbf{y}_{i-1} + \mathbf{b}_i) \quad (5)$$

where i denotes the layer number, \mathbf{y}_{i-1} stands for the output of layer $i - 1$, \mathbf{W}_i and \mathbf{b}_i denotes the weights and bias at layer i respectively, and α_i the activation function.

The decoder function $f_D(\cdot)$, also a composed function as $f_E(\cdot)$ in eq. 4, maps the code to an estimated reconstruction of the input vector by

$$\hat{\mathbf{x}} = f_D(f_E(\tilde{\mathbf{x}}_c)) \quad (6)$$

where $\hat{\mathbf{x}}$ is the approximation of the input vector, updating the weights \mathbf{W}_i and bias \mathbf{b}_i by backpropagating the loss d computed by a loss function L as

$$d = L(\tilde{\mathbf{x}}_c, f_D(f_E(\tilde{\mathbf{x}}_c))) = L(\tilde{\mathbf{x}}_c, \hat{\mathbf{x}}) \quad (7)$$

where L computes the distance between $\tilde{\mathbf{x}}_c$ and $\hat{\mathbf{x}}$, and \mathbf{W}_i and \mathbf{b}_i are updated approaching $d \rightarrow 0$, by an optimization algorithm such as stochastic gradient descent, Adam or Adagrad.

3.2.3. Representations generation (Inference)

Given the C trained AEs, denoted as f_c , the set of cloud free observations of individual geographic points (pixels) form an array $\tilde{\mathbf{X}} \in \mathbb{R}^{t \times F}$, where t denotes the number of cloud free samples for a given pixel. $\tilde{\mathbf{X}}$ is the input to all the AEs, and the reconstruction array is obtained from each AE as

$$\hat{\mathbf{X}}_c = f_c(\tilde{\mathbf{X}}) \quad (8)$$

where $\hat{\mathbf{X}}_c$ represents the reconstruction estimated by AE c . Then, the difference vector is computed by the same loss function used in training phase (eq. 7) as

$$\mathbf{D}_c = L(\tilde{\mathbf{X}}, \hat{\mathbf{X}}_c) \quad (9)$$

and the pixel mean reconstruction difference vector $\bar{\mathbf{d}}_c$ is computed by

$$\bar{\mathbf{d}}_c = \frac{1}{t} \sum_{s=0}^t \mathbf{d}_{sc} \quad (10)$$

where \mathbf{d}_{sc} denotes the s -th row of \mathbf{D}_c and the representations are formed by concatenating the mean pixel reconstruction difference vector as

$$\mathbf{z} = \bar{\mathbf{d}}_1 \oplus \bar{\mathbf{d}}_2 \oplus \cdots \oplus \bar{\mathbf{d}}_C \quad (11)$$

where \oplus denotes the vector concatenation and $\bar{\mathbf{d}}_c$ the mean pixel reconstruction difference vector from AE c . The inference phase of our proposed framework is presented in Figure 8

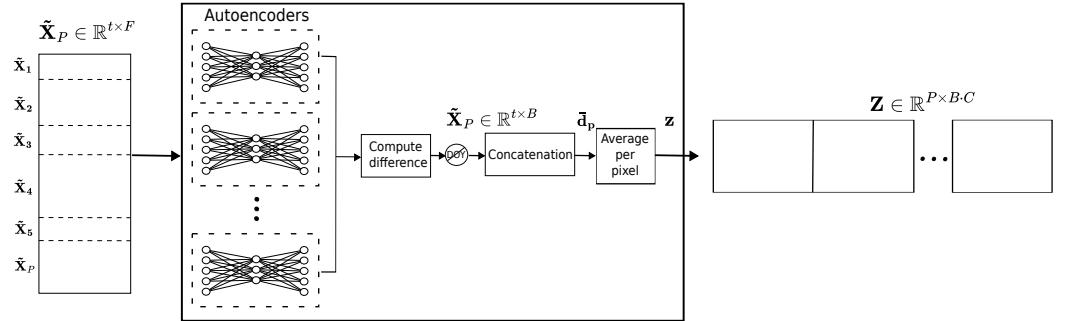


Figure 8. Inference workflow of the proposed framework. For each temporal set of cloud-free reflectance spectra, the average reconstruction difference vector are calculated for each of the C autoencoders (AEs) and concatenated to define the representations of this pixel.

4. Experimental results

4.1. Dataset

For this work the public benchmark dataset *Breizhcrops* presented in [12] was used for experiments and evaluation. This dataset is available at the GitHub repository (<https://github.com/dl4sits/breizhcrops>). The provided multi-temporal multi-spectra data is from the Brittany region in the northwest of France and is composed of labeled Sentinel-2 images from January 1st to December 31st, 2017. Labels are assigned to the "average of reflectance values over the bounds of the field geometry retrieved from the dataset" [12].

This dataset is organized in four regions (see Table 2), and each region contains nine crop categories: barley, wheat, rapeseed, corn, sunflower, orchards, nuts, permanent meadows and temporary meadows. To allow a direct comparison to the work published in [12], we use the regions FRH01 and FRH02 for training, FRH03 for validation, and FRH04 for evaluation.

Table 2. Regions of Brittany (France) with number of field parcels and spectral data for the atmospherically corrected surface reflectances at the bottom-of-atmosphere (L2A) [12]. The regions FRH01 and FRH02 were used for training, FRH03 for validation, and FRH04 for evaluation.

Regions	NUTS-3	L2A
Côtes-d'Armor	FRH01	178,632
Finistère	FRH02	140,782
Ille-et-Vilaine	FRH03	166,367
Morbihan	FRH04	122,708
Total		608,489

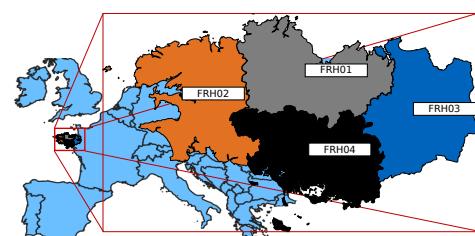


Table 3 describes the number of samples per class used for training, validation and test respectively. It is worth noting that the dataset is highly imbalanced; the most abundant

class "temporary meadows has > 300 times samples compared to the two minor classes "sunflower" and "nuts". This makes the classification model more sensitive to overfitting and also makes an accuracy evaluation more difficult [67].

Table 3. Number of samples per class used for training, validation and test.

Class	Training	Validation	Test	Total
Barley	23,787	7,154	5,981	36,922
Wheat	45,406	27,202	17,009	89,617
Rapeseed	7,945	3,557	3,244	14,746
Corn	80,623	42,011	31,361	153,995
Sunflower	7	10	2	19
Orchards	1,285	1,217	552	3,054
Nuts	28	10	11	49
Perm. Meadows	69,177	32,524	25,134	126,835
Temp. Meadows	91,156	52,682	38,414	182,252

It is worth mentioning that this dataset provides only spectral signatures in tabular format for the center pixel in a field and not Sentinel 2 images.

4.2. AEs training

With the aim of developing an algorithm capable to process relatively large geographic areas, the AE is composed by a single layer FCN as encoder, and its counterpart for the decoder. This keeps computational load and processing times lower than other models such as convolutional or recurrent networks. Batch size, learning rate, number of units in hidden layer and the loss function were set in accordance with the results acquired through the hyperparameter random search presented in Appendix A. The split of the dataset is described in Table 4, which outlines the features employed in this experiment. These include DOY (sine and cosine), 10 spectral bands (10 and 20 meters resampled to 10m) and five well-known spectral indices (NDWI, NDVI, NDTI, NDSVI and EVI). Table 5 presents the AEs configuration.

Table 4. Training, validation and testing split, and number of input features and classes considered for the autoencoders.

Parameter	Value
Training size	319,414
Validation size	166,367
Testing size	122,708
Features	10 bands, 2 DOY, 5 spectral indices
Classes	9

Table 5. Autoencoders (AEs) hyperparameters final configuration established by random search.

Hyperparameter	Value
Epochs	10000
Early stop	True
Patience	10
Min. delta	1e-5
Batch size rate	0.05*
Units in hidden layers	5
Learning rate	1e-4
Optimizer	Adam
Loss	MSE

* proportion of samples for each class

4.3. Separability assessment and distance metrics

For qualitative assessment of the inter-class separability in the generated representation space, 3D scatterplots of the test spectral-temporal Sentinel-2 BOA data and their corresponding representations produced by our method, reduced to a three-dimensional space by t-distributed Stochastic Neighbor Embedding (TSNE), are shown in Figures 9a and 9b respectively. The scatter plot of representations, compared with that of the initial data, shows that the density of points belonging to each of the crop types is much better clustered.

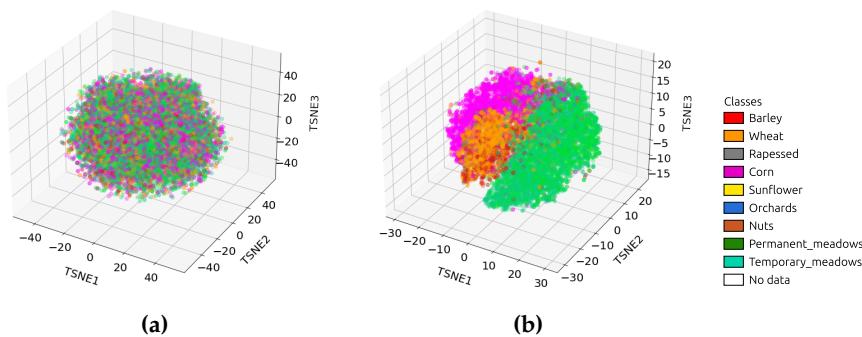


Figure 9. 3D-scatterplot of (a) S2 fixed-length time series (45 observations) and (b) representation, over three principal components obtained by t-distributed Stochastic Neighbor Embedding (TSNE) only for visual interpretation.

Several distance metrics were used to assess the distance between classes. Table 6 presents a comparison of the inter-class separability on the input spectral-temporal Sentinel-2 BOA data and the generated representations, measured by: Silhouette score (*SS*), which ranges from -1 for incorrect clustering and +1 for highly dense clustering, Calinski-Harabasz index (*CH*), for which larger scores indicates better separability, and Davies-Bouldin index (*DBI*), which ranges from 0 to ∞ and the closer to zero the better the separability between clusters (see Appendix B for metrics definitions).

In the same way as in the qualitative analysis, distance scores demonstrate much higher separability on the representation space than on the initial data.

Table 6. Class distance assessment of the S2 dataset and the representations produced by our method. Silhouette score (*SS*), Calinski-Harabasz index (*CH*), Davies-Bouldin index (*DBI*).

Distance metric	S2 data	Our approach
<i>SS</i>	-0.76	0.20
<i>CH</i>	1.4	73074.46
<i>DBI</i>	72.44	18.96

4.4. Evaluating representations in the classification of crop types

Once the representations have been produced, a 3-layer FCN was used as classification model, where the inputs are the generated representations and their corresponding labels. The parameters of the classifier are detailed in Table 7. The elements corresponding to the differences in the DOY embeddings are not considered as part of the final representation vectors, since averaging these elements induces redundancy to the classification model. Hence, the input dimensionality is defined by B spectral bands plus 5 spectral indices times C number of AEs.

Table 7. Classification model configuration.

Hyperparameter	Value
Input size	$(B + 5) \times C = 135$
Epochs	10000
Batch size	1000
Units in hidden layers	128, 64, 32
Learning rate	1e-4
Optimizer	Adam
Loss	Categorical crossentropy

A comparative study is presented in Table 8 where the performance of different traditional classifiers, such as RF, SVM, XGBoost, and FCN, are evaluated using two types of input data: fixed length Sentinel-2 BOA data and our derived representations. The evaluation is based on overall accuracy (OA), Cohen's kappa coefficient (κ) and Matthews

correlation coefficient (MCC). All models are tested with exactly the same training and testing samples of the BreizhCrops dataset (as described in Table 3) allowing a direct comparison of results.

In all cases, our representations improved classification performance compared to the use of the original Sentinel-2 data. Improvement ranged from 1% to 4% in terms of OA. The FCN classifier achieves the best OA, κ and MCC with both input types, and stands out particularly using the representations as input, reaching the highest OA=0.767, κ =0.695 and MCC=0.698.

Table 8. Comparison of classification performance with Sentinel-2 data versus representations produced by our method as input data to conventional classifiers, random forest (RF), support vector machine (SVM), extreme gradient boosting (XGBoost) and fully connected network (FCN) evaluated by overall accuracy (OA), Cohen's kappa coefficient (κ), and Matthews correlation coefficient (MCC).

Metric	S2 data				Representations			
	RF	SVM	XGBoost	FCN	RF	SVM	XGBoost	FCN
OA	0.7172	0.7091	0.7036	0.7438	0.7345	0.7466	0.7139	0.7672
κ	0.6264	0.6197	0.6113	0.6716	0.6464	0.6688	0.6201	0.6953
MCC	0.6326	0.6249	0.6149	0.6729	0.6491	0.6710	0.6234	0.6977

We conducted a comparison between representations and S2 data across all classifiers in scenarios with limited labeled data. Results presented in Figure 10 demonstrate that the representations consistently offer greater stability and maintain higher accuracy as the percentage of available training data decreases.

The FCN benefits exceptionally from representations, it obtains the highest OA of ~ 0.77 using 100% of the training data, and its classification performance is much less affected compared to the use of the original Sentinel-2 data as the number of training samples decreases. SVM, in combination with the representations, keeps the classification performance stable, even with very low number of training samples (around 30,000), offering clear improvements over S2 data and making it the best option in this scenario. XGBoost and RF, on the other hand, offer stable performance with S2, but the use of representations considerably elevates accuracy.

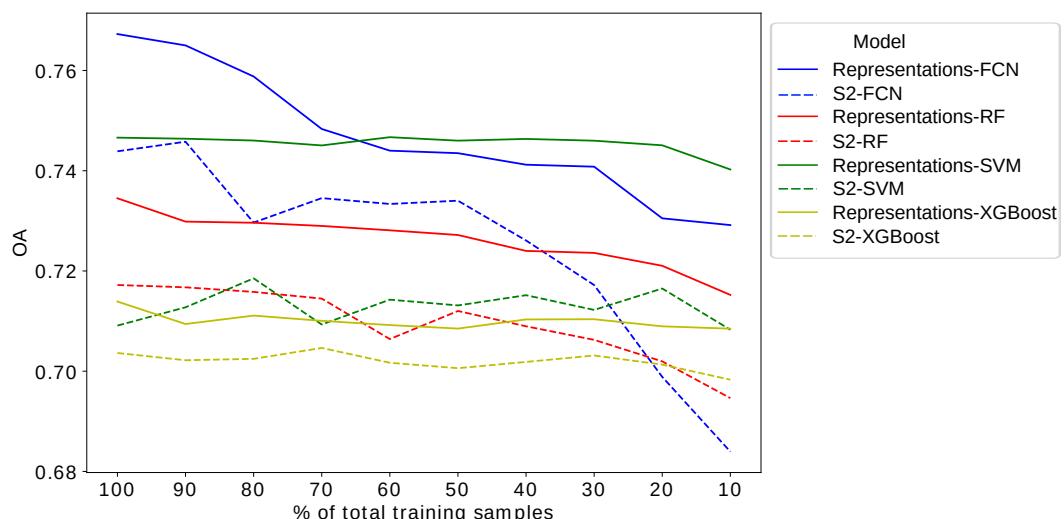


Figure 10. Overall Accuracy (OA) of random forest (RF), support vector machine (SVM), extreme gradient boosting (XGBoost) and fully connected network (FCN) trained with variable percentage of training samples and using (i) representations (solid line), and (ii) original Sentinel-2 data (broken line).

The confusion matrix shown in Table 9 presents the model performance on the testing dataset. Our approach performs relatively accurate for wheat, rapeseed and corn samples,

and, although permanent and temporary meadows samples are not accurately classified, both classes are actually the same crop type, and misclassified samples are mainly due to their similar nature as shown in the last two rows of the confusion matrix. However, our approach is inaccurate for the small and challenging classes in this dataset, i.e., sunflower, orchards and nuts. The misclassification of these classes is probably not related to incapacity of the model to deal with imbalanced datasets, but is a direct result of the very limited number of samples.

Table 9. Confusion matrix of the fully connected network (FCN) prediction for the testing data.

Object based	Barley	Wheat	Rapeseed	Corn	Sunflower	Orchards	Nuts	Permanent meadows	Temporary meadows
Barley	4653	760	40	163	0	0	0	31	334
Wheat	535	15850	14	179	0	0	0	79	352
Rapeseed	208	25	2864	42	0	0	0	13	92
Corn	267	422	22	29547	0	0	0	104	999
Sunflower	0	0	0	0	0	0	0	1	1
Orchards	3	6	0	5	0	0	0	290	248
Nuts	0	0	0	2	0	0	0	5	4
Permanent meadows	287	246	14	164	0	0	0	11985	13438
Temporary meadows	562	439	53	568	0	0	0	7541	29251

Table 10 presents a performance comparison of our method with convolutional, recurrence and attention-based methods. TempCNN, OmnicSCNN, LSTM, StarRNN, Transformer, and our proposed method with the generated representations as input to a FCN are evaluated by OA, average precision (AP), F1 score, and κ . Additionally, details on the processor used, the number of parameters, and runtime (it/s) are presented. We present the results reported in [12], since the code used for those experiments is not publicly available and it is not straightforwardly reproducible. Notwithstanding, the same test data points were used in our experiments, hence, results are directly comparable.

Table 10. Classification performance evaluation of benchmarked models by overall accuracy (OA), average precision (AP), F1 score (F1) and Cohen's kappa coefficient (κ). All models were evaluated over the same testing dataset.

	TempCNN	OmniscCNN	LSTM	StartRNN	Transformer	AE-FCN
OA	0.79	0.73	0.80	0.79	0.80	0.77
AP	0.55	0.52	0.57	0.56	0.58	0.54
F1	0.79	0.72	0.80	0.79	0.80	0.76
κ	0.73	0.65	0.74	0.73	0.75	0.70
Processor	8X NVIDIA Tesla P100 16GB/GPU 28,672 Total NVIDIA CUDA Cores					
Nº param	3,199,501	2,739,737	1,339,431	72,103	188,429	6,825
Runtime in [it/s]	1.25	1.02	1.16	1.02	1.20	0.75

Our representations-based approach combined with the FCN is substantially less computationally expensive than the other benchmarked methods, requiring only 6,825 trainable parameters, compared to the 1,338,431 for LSTM and 188,429 for Transformer. This means our method uses roughly 200 and 28 times fewer parameters, respectively, compared to the two deep learning methods. This reduction directly impacts computational load and consequently processing time, as seen in the runtime of 0.75 it/s for FCN, which, despite running on much lower capacity hardware, is just slightly slower than Transformer (1.20 it/s) and other sophisticated models such as TempCNN (1.25 it/s) and OmnicSCNN (1.02 it/s).

In addition to this substantial reduction of trainable parameters, our method maintains competitive classification accuracy, achieving an OA of 0.767, just 3% lower than

Transformer's OA of 0.80. Furthermore, the accuracy of our method is comparable to TempCNN (OA=0.79) and OmniscCNN (OA=0.73). However, these two approaches are computationally far more expensive, requiring approximately 400 times more parameters than AE-derived representations within a simple FCN (AE-FCN).

In terms of metrics that weight class imbalance, Transformer has the highest AP of 0.58, while AE-FCN obtains a competitive 0.54. LSTM and Transformer obtained the highest F1 score (both 0.80), whereas AE-FCN achieves a competitive 0.76, indicating a balanced performance despite having far fewer parameters. Similarly, LSTM and Transformer have the highest κ score (0.75), with AE-FCN scoring 0.70, which is penalized owing to mismatches in particular classes but remains effective given the model's simplicity.

4.5. Qualitative results

To enable a qualitative analysis over a contiguous spatial extent and not simply on a tabulated dataset, 67 Sentinel-2 multispectral images from 2017 of a subregion in FRH04 (test region) were downloaded and preprocessed. A representative area was defined drawing a polygon where most of the classes (barley, wheat, corn, rapeseed, temporary meadows and permanent meadows) are present (Figure 11a).

Representations for this study area are produced by passing individual pixels from the imagery dataset through the inference workflow outlined in Figure 8. Exemplary composite images generated by combining three random representations are presented in Figures 11b, 11c, 11d, 11e, 11f, 11g and 11h, which clearly contrast the crop fields in the new representation space.

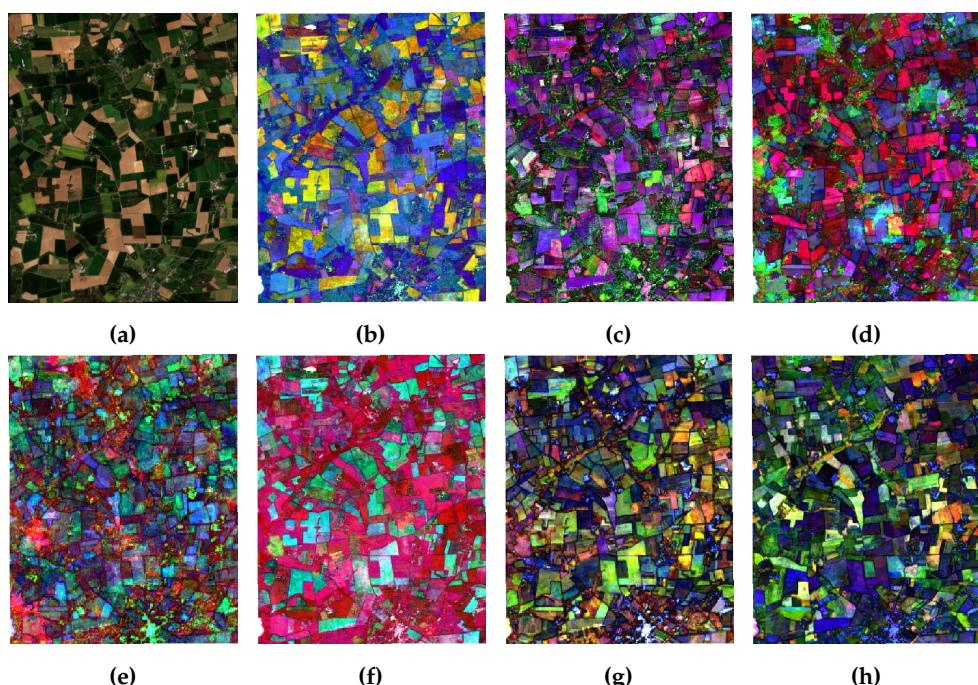


Figure 11. (a) True color image of the study area in 2017 and composites images generated by combining three random representations per map (b) 9-64-30, (c) 59-84-81, (d) 30-11-141, (e) 45-66-57, (f) 20-10-32, (g) 5-142-83 and (h) 24-79-133.

A classification map produced by our method is presented in Figure 12. Figure 12b illustrates a pixel-based classification, i.e., without considering field boundaries or spatial context. Misclassifications are mainly seen near field edges, since these are not pure pixels and often contain mixed spectral data.

To better illustrate the potential of our method in real-world activities, Figure 12c presents a field-based classification map, where the output of our method is post-processed to group the pixel-wise predictions into polygon-level prediction by computing the mode

of predictions within the field borders. This map preserves field structure, creating a more coherent and interpretable map. The strong similarity between Figures 12a and 12c show that the representations are sufficiently representative for crop type classification.

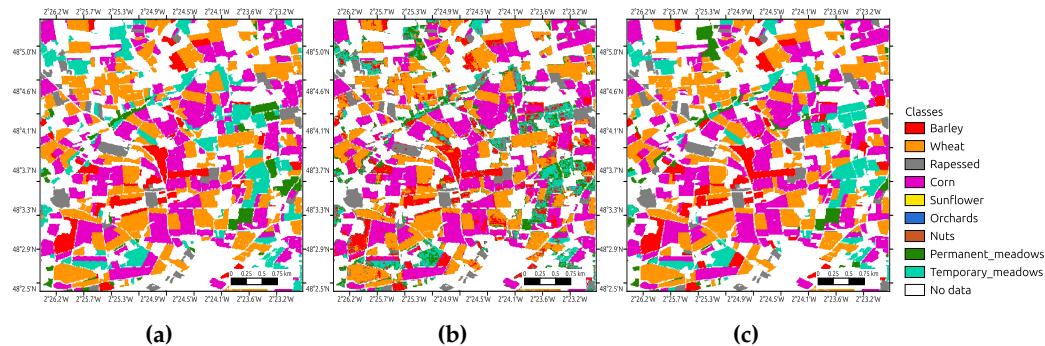


Figure 12. (a) Study area ground truth at field level (polygons), (b) representations-based fully connected network (FCN) pixel-wise classification (raster) and (c) representations-based FCN field-based classification (polygons).

As reported in the confusion matrix (Table 9), qualitative results illustrate that temporary meadow fields are frequently confused with permanent meadows. However, as discussed in subsection 4.4, these two crop type share similar spectral signatures, particularly when observed during different seasons of the year. This spectral overlap makes it challenging for classifiers to distinguish between these two crop types.

5. Discussion

Our AEs-based methodology for RL addresses the problem of cloud-corrupted optical data by mapping RS spectral-temporal features into informative, and gap free representations. Our spectral and temporal-based approach produces pixel level comprehensive representations, while avoiding the need to employ complex spatial-based classifiers.

The method proposed in this paper has as main advantage its ability to produce pixel-wise representations independently of the number of cloud free samples. Therefore, complex interpolation/gap filling methods, used in other approaches, are not needed. Other solutions, such as obtaining fixed-length time series matching with the input size of a neural network are not needed. The derived representations from our method will be gap free as long as at least one valid observation is present in the time series. Obviously, the stability of the representations increase with increasing number of valid observations.

Despite the restricted depth of our method for, both (i) the representations learning process, as well as for (ii) the classifier, our method performs satisfactory extracting meaningful information for downstream crop classification. While some deep classification networks attain slightly better OA scores, our lightweight model performs similarly with substantially less complexity. Importantly, the use of representations instead of the Sentinel-2 data improves performance for all evaluated models, while maintaining low computational load. The usefulness of the representation increased compared to the original data, when fewer training samples were available for training.

Models, such as TempCNN, OmniscCNN, LSTM, StarRNN, and Transformer, need to be executed on powerful equipment well-suited for handling complex models. In contrast, our full framework was easily launched on a significantly less powerful CPU. This showcases our method's efficiency and adaptability to lower-end hardware and/or scalability to large geographic areas. In terms of number of trainable parameters, convolutional and recurrent models require millions of parameters, which indicates their high computational demands.

The dataset used in the experiments of this work is particularly challenging, as sunflower, orchards, and nuts were not separable by any of the analyzed algorithms, mainly due to the limited number of labeled samples. The few available samples restricted our

model from learning enough informative and significative representations before classification. As other methods, our approach is still negatively affected in classification performance when extraordinarily limited number of samples are available. However, there are no computational or data limitations to apply our approach on different areas, and a simple solution is to slightly expand labeled data collection, even using samples from other already labeled regions. In addition, our approach has no computational or data limitations to be executed on different areas and with other optical sensors datasets, such as Landsat, and even from radar sensors. Work is underway to see how classification performance changes if the AEs are trained without focusing on specific classes. If successful, this would yield a fully self-supervised learning algorithm for representation learning.

6. Conclusions

Based on the results reported in this paper, we draw the following conclusions:

1. Quantitative evaluations based on various distance metrics demonstrate that the representations produced by our method accomplished the objective of mapping cloud-corrupted irregular RS spectral-temporal data to a feature space where inter-class separability is higher than in the initial Sentinel 2 BOA time series.
2. Classification scores achieved by our method, alongside the comparison of trainable parameters and execution time, highlight the efficiency of our method. While our model delivers slightly lower OA compared to more complex methods such as LSTM, it outperforms conventional classifiers in terms of accuracy, all with significantly reduced computational load. This makes it highly scalable to larger areas without excessive time consumption, offering an effective balance between performance and efficiency.
3. Our method performs correctly for the majority of classes evaluated in this work, especially for those with sufficient training samples. When only few training samples were available, our method showed the same problems as the baseline methods.

In summary, experimental results demonstrate that this work successfully introduce a novel RL method for crop type classification and confirm the main characteristics of our method: 1) scalability; i.e. ability to process large areas of interest, as it is often required in real-world activities, 2) input length flexibility and no reliance on gap filling methods, 3) competitive trade-off between computational demands and classification performance, and 4) direct applicability for other downstream tasks.

Outside the scope of this work, there are still some points to consider in future research:

- Implementation of a fully unsupervised methodology for training autoencoders without relying on labeled data.
- Evaluation of the proposed methodology on other optical sensors, radar sensors or combination of several sensor modalities.
- Fine-tuning the RL-classification model to find a better balance between performance metrics and number of trainable parameters.
- Although this research presents specifically crop type classification task-guided representations, the extrapolation to other classification task is straightforward.

Author Contributions: Conceptualization, A.G. and C.A.; methodology, C.A., A.G. and J.L.; software, A.G. and J.L.; writing—original draft preparation, A.G. and J.L.; writing—review and editing, C.A. and D.T.; visualization, A.G.; supervision, C.A. and D.T.; project administration, A.G.; funding acquisition, D.T.

Funding: This research was funded by CONAHCYT grant number 1001207.

Conflicts of Interest: The authors declare no conflict of interest

Abbreviations

The following abbreviations are used in this manuscript:

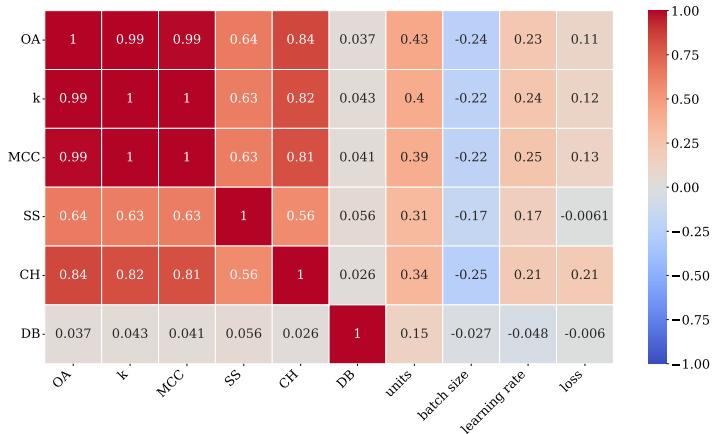
AEs	Autoencoders	537
ANN	Artificial neural networks	538
AP	Average precision	
BOA	Bottom of atmosphere	
CH	Calinski harabasz	
DBI	Davies bouldin index	
DOY	Day-of-year	
DT	Decision trees	
FCN	Fully connected network	
GEE	Google earth engine	
κ	Cohen's kappa coefficient	
KS	Kennard-Stone	
LSTM	Long-short term memory	
MCC	Matthews correlation coefficient	
ML	Machine learning	539
MSE	Mean square error	
OA	Overall accuracy	
PCA	Principal component analysis	
RF	Random forest	
RL	Representation learning	
ROI	Region of interest	
RS	Remote sensing	
SS	Silhouette score	
STBT	Spectral-temporal Barlow twins	
SVM	Support vector machine	
TOA	Top of atmosphere	
UA	User's accuracy	
XGBoost	Extreme gradient boosting	

Appendix A Hyperparameters random search

AEs hyperparameters were defined after an extensive random search. One hundred configurations with four variable hyperparameters were launched and evaluated with three classification and three distance metrics. The search spaces for each hyperparameter are:

- Units: $U\{1, 16\}$
- Batch size rate: $U[0.1, 0.3]$
- Learning rate: $U[1 \times 10^{-3}, 9 \times 10^{-6}]$
- Loss: $\{0, 1\}$

where $U\{\cdot\}$ and $U[\cdot]$ denote uniform discrete and continuous distribution respectively. Final configuration reported in Table 5 was defined according to the pairwise correlation between hyperparameters and metrics presented in Figure A1.

**Figure A1.** Hyperparameters and quality indicators correlation matrix.**Appendix B Separability metrics**

These metrics quantify how separable a set of classes/clusters are from each other. 552

Silhouette score:

$$SS = \frac{b - a}{\max(a, b)} \quad (A1)$$

where a is the mean distance between a sample and all other points in the same class, b is the mean distance between a sample and all other points in the next nearest cluster. The score is bounded between -1 for incorrect clustering and +1 for highly dense clustering. 555
556
557
558
559

Calinski-Harabasz Index

$$CH = \frac{\left[\frac{\sum_{k=1}^K n_k \|c_k - c\|^2}{K-1} \right]}{\left[\frac{\sum_{k=1}^K \sum_{i=1}^{n_k} \|d_i - c_k\|^2}{N-K} \right]} \quad (A2)$$

where d_i is the feature vector of data point i , n_k is the size of the k^{th} cluster, c_k is the feature vector of the centroid of the k^{th} cluster, c is the feature vector of the global centroid of the entire dataset, N is the total number of data points. The higher the score is the better separation. 560
561
562
563
564
565
566

Davies-Bouldin Index

$$R_{ij} = \frac{s_i + s_j}{d_{ij}} \quad (A3)$$

$$DBI = \frac{1}{k} \sum_{i=1}^k \max_{i \neq j} R_{ij} \quad (A4)$$

where s_i is the average distance between each point of cluster i and the centroid of that cluster, d_{ij} is the distance between cluster centroids i and j . The score is between 0 and ∞ , and the values closer to zero indicate a better separation. 567
568
569
570
571

Appendix C Classification metrics

For evaluating the predictions obtained to the FCN, we consider to compute the same metrics that the authors in [12] used for comparative purposes of this work. We compute through of the confusion matrix the equations shown follow: 572
573
574
575

Given a confusion matrix $\mathbf{M} \in \mathbb{R}^{C \times C}$ where C is the number of classes, the OA is computed with the equation A5.

$$OA = \frac{\sum_{i=1}^C \mathbf{M}_{ii}}{\sum_{i=1}^C \sum_{j=1}^C \mathbf{M}_{ij}} \quad (\text{A5})$$

From \mathbf{M} to a class-wise confusion matrix following the approach one versus all, the producers accuracy (PA) also known as precision is computed by

$$PA_c = \frac{TP_c}{TP_c + FP_c} \quad (\text{A6})$$

where TP_c is the true positive and FP_c is the false positive of the class c .

Then, the average precision (AP) is computed as follows

$$AP = \frac{\sum_{c=1}^C PA_c}{C} \quad (\text{A7})$$

The user's accuracy (UA_c) also known as recall is compute as follows

$$UA_c = \frac{TP_c}{TP_c + FN_c} \quad (\text{A8})$$

where TP_c is the true positive and FN_c is the false negative of the class c .

With the equation A6 and A8 we can compute the F1-score per class ($F1_c$) as follows:

$$F1_c = 2 \frac{PA_c \times UA_c}{PA_c + UA_c} \quad (\text{A9})$$

and the weighted F1-score is computed by

$$F1 = \sum_{c=1}^C w_c \times F1_c \quad (\text{A10})$$

where $w_c = \frac{N_c}{N}$ and N_c is the number of samples in class c and N denotes the total number of samples.

The formula for Cohen's kappa coefficient (κ) is the probability of agreement minus the probability of random agreement, divided by one minus the probability of random agreement.

$$\kappa = \frac{p_o - p_e}{1 - p_e} \quad (\text{A11})$$

where p_o is is the relative observed agreement among raters, and p_e is the hypothetical probability of chance agreement.

The multiclass Matthew's correlation coefficient (MCC) is defined by

$$MCC = \frac{cp \times s - \sum_c p_c \times t_c}{\sqrt{(s^2 - \sum_c p_c^2) \times (s^2 - \sum_c t_c^2)}} \quad (\text{A12})$$

where $t_c = \sum_i^C \mathbf{M}_{ic}$ represents the number of times that class c really happened., $p_c = \sum_i^C \mathbf{M}_{ci}$ denotes the number of times class c has been predicted , $cp = \sum_c^C \mathbf{M}_{cc}$ indicates the number of samples that have been correctly predicted and $s = \sum_i^C \sum_j^C \mathbf{M}_{ij}$ is the overall number of samples.

References

1. Vuolo, F.; Neuwirth, M.; Immitzer, M.; Atzberger, C.; Ng, W.T. How much does multi-temporal Sentinel-2 data improve crop type classification? *Int. J. Appl. Earth Obs.* **2018**, *72*, 122–130. <https://doi.org/10.1016/j.jag.2018.06.007>. 601
2. Pelletier, C.; Webb, G.; Petitjean, F. Temporal Convolutional Neural Network for the Classification of Satellite Image Time Series. *Remote Sensing* **2019**, *11*, 523. <https://doi.org/10.3390/rs11050523>. 603
3. Foerster, S.; Kaden, K.; Foerster, M.; Itzerott, S. Crop type mapping using spectral-temporal profiles and phenological information. *Comput. Electron. Agr.* **2012**, *89*, 30–40. <https://doi.org/10.1016/j.compag.2012.07.015>. 605
4. Chen, B.; Zheng, H.; Wang, L.; Hellwich, O.; Chen, C.; Yang, L.; Liu, T.; Luo, G.; Bao, A.; Chen, X. A joint learning Im-BiLSTM model for incomplete time-series Sentinel-2A data imputation and crop classification. *Int. J. Appl. Earth Obs.* **2022**, *108*, 102762. <https://doi.org/10.1016/j.jag.2022.102762>. 609
5. Tariq, A.; Yan, J.; Gagnon, A.S.; Riaz Khan, M.; Mumtaz, F. Mapping of cropland, cropping patterns and crop types by combining optical remote sensing images with decision tree classifier and random forest. *Geo-spatial Information Science* **2022**, *26*, 302–320. <https://doi.org/10.1080/10095020.2022.2100287>. 610
6. Gao, F.; Zhang, X. Mapping Crop Phenology in Near Real-Time Using Satellite Remote Sensing: Challenges and Opportunities. *Journal of Remote Sensing* **2021**, *2021*. <https://doi.org/10.34133/2021/8379391>. 614
7. Palchowdhuri, Y.; Valcarce-Díñeiro, R.; King, P.; Sanabria-Soto, M. Classification of multi-temporal spectral indices for crop type mapping: a case study in Coalville, UK. *The Journal of Agricultural Science* **2018**, *156*, 24–36. <https://doi.org/10.1017/s0021859617000879>. 616
8. Heupel, K.; Spengler, D.; Itzerott, S. A Progressive Crop-Type Classification Using Multitemporal Remote Sensing Data and Phenological Information. *PFG – Journal of Photogrammetry, Remote Sensing and Geoinformation Science* **2018**, *86*, 53–69. <https://doi.org/10.1007/s41064-018-0050-7>. 619
9. Li, Q.; Tian, J.; Tian, Q. Deep Learning Application for Crop Classification via Multi-Temporal Remote Sensing Images. *Agriculture-london* **2023**, *13*, 906. <https://doi.org/10.3390/agriculture13040906>. 621
10. Dong, H.; Ma, W.; Wu, Y.; Zhang, J.; Jiao, L. Self-Supervised Representation Learning for Remote Sensing Image Change Detection Based on Temporal Prediction. *Remote Sensing* **2020**, *12*, 1868. <https://doi.org/10.3390/rs12111868>. 624
11. Paris, C.; Weikmann, G.; Bruzzone, L. Monitoring of agricultural areas by using Sentinel 2 image time series and deep learning techniques. In Proceedings of the Image and Signal Processing for Remote Sensing XXVI; Notarnicola, C.; Bovenga, F.; Bruzzone, L.; Bovolo, F.; Benediktsson, J.A.; Santi, E.; Pierdicca, N., Eds. SPIE, 2020. <https://doi.org/10.1117/12.2574745>. 626
12. Rußwurm, M.; Pelletier, C.; Zollner, M.; Lefèvre, S.; Körner, M. BREIZHCROPS: A TIME SERIES DATASET FOR CROP TYPE MAPPING. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* **2020**, *XLIII-B2-2020*, 1545–1551. <https://doi.org/10.5194/isprs-archives-xliii-b2-2020-1545-2020>. 629
13. Nowakowski, A.; Mrziglod, J.; Spiller, D.; Bonifacio, R.; Ferrari, I.; Mathieu, P.P.; Garcia-Herranz, M.; Kim, D.H. Crop type mapping by using transfer learning. *International Journal of Applied Earth Observation and Geoinformation* **2021**, *98*, 102313. <https://doi.org/10.1016/j.jag.2021.102313>. 632
14. Gadiraju, K.K.; Vatsavai, R.R. Remote Sensing Based Crop Type Classification Via Deep Transfer Learning. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* **2023**, *16*, 4699–4712. <https://doi.org/10.1109/jstars.2023.3270141>. 635
15. Wu, B.; Zhang, M.; Zeng, H.; Tian, F.; Potgieter, A.B.; Qin, X.; Yan, N.; Chang, S.; Zhao, Y.; Dong, Q.; et al. Challenges and opportunities in remote sensing-based crop monitoring: a review. *National Science Review* **2022**, *10*. <https://doi.org/10.1093/nsr/nwac290>. 637
16. Yin, L.; You, N.; Zhang, G.; Huang, J.; Dong, J. Optimizing Feature Selection of Individual Crop Types for Improved Crop Mapping. *Remote Sensing* **2020**, *12*, 162. <https://doi.org/10.3390/rs12010162>. 640
17. Yi, Z.; Jia, L.; Chen, Q. Crop Classification Using Multi-Temporal Sentinel-2 Data in the Shiyang River Basin of China. *Remote Sensing* **2020**, *12*, 4052. <https://doi.org/10.3390/rs12244052>. 641
18. He, S.; Peng, P.; Chen, Y.; Wang, X. Multi-Crop Classification Using Feature Selection-Coupled Machine Learning Classifiers Based on Spectral, Textural and Environmental Features. *Remote Sensing* **2022**, *14*, 3153. <https://doi.org/10.3390/rs14133153>. 644
19. Dumeur, I.; Valero, S.; Inglada, J. Self-Supervised Spatio-Temporal Representation Learning of Satellite Image Time Series. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2024**, *17*, 4350–4367. <https://doi.org/10.1109/jstars.2024.3358066>. 646
20. Wang, S.; Azzari, G.; Lobell, D.B. Crop type mapping without field-level labels: Random forest transfer and unsupervised clustering techniques. *Remote Sens. Environ.* **2019**, *222*, 303–317. <https://doi.org/10.1016/j.rse.2018.12.026>. 649
21. Maponya, M.G.; van Niekerk, A.; Mashimbye, Z.E. Pre-harvest classification of crop types using a Sentinel-2 time-series and machine learning. *Computers and Electronics in Agriculture* **2020**, *169*, 105164. <https://doi.org/10.1016/j.compag.2019.105164>. 650
22. Hu, Q.; Wu, W.; Song, Q.; Yu, Q.; Lu, M.; Yang, P.; Tang, H.; Long, Y. Extending the Pairwise Separability Index for Multicrop Identification Using Time-Series MODIS Images. *IEEE Trans. Geosci. Remote Sens.* **2016**, *54*, 6349–6361. <https://doi.org/10.1109/tgrs.2016.2581210>. 652
23. Roy, D.; Yan, L. Robust Landsat-based crop time series modelling. *Remote Sens. Environ.* **2020**, *238*, 110810. <https://doi.org/10.1016/j.rse.2018.06.038>. 655
24. Kussul, N.; Lavreniuk, M.; Skakun, S.; Shelestov, A. Deep Learning Classification of Land Cover and Crop Types Using Remote Sensing Data. *IEEE Geosci. Remote Sens. Lett.* **2017**, *14*, 778–782. <https://doi.org/10.1109/lgrs.2017.2681128>. 657

25. Ingla, J.; Vincent, A.; Arias, M.; Tardy, B.; Morin, D.; Rodes, I. Operational High Resolution Land Cover Map Production at the Country Scale Using Satellite Image Time Series. *Remote Sensing* **2017**, *9*, 95. <https://doi.org/10.3390/rs9010095>. 659
660
26. Cai, Y.; Guan, K.; Peng, J.; Wang, S.; Seifert, C.; Wardlow, B.; Li, Z. A high-performance and in-season classification system of field-level crop types using time-series Landsat data and a machine learning approach. *Remote Sens. Environ.* **2018**, *210*, 35–47. <https://doi.org/10.1016/j.rse.2018.02.045>. 661
662
663
27. Feng, S.; Zhao, J.; Liu, T.; Zhang, H.; Zhang, Z.; Guo, X. Crop Type Identification and Mapping Using Machine Learning Algorithms and Sentinel-2 Time Series Data. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2019**, *12*, 3295–3306. <https://doi.org/10.1109/jstars.2019.2922469>. 664
665
666
28. Zhong, L.; Hu, L.; Zhou, H. Deep learning based multi-temporal crop classification. *Remote Sensing of Environment* **2019**, *221*, 430–443. <https://doi.org/10.1016/j.rse.2018.11.032>. 667
668
29. Prins, A.J.; Van Niekerk, A. Crop type mapping using LiDAR, Sentinel-2 and aerial imagery with machine learning algorithms. *Geo-spatial Information Science* **2020**, *24*, 215–227. <https://doi.org/10.1080/10095020.2020.1782776>. 669
670
30. Manish Lad, A.; Mani Bharathi, K.; Akash Saravanan, B.; Karthik, R. Factors affecting agriculture and estimation of crop yield using supervised learning algorithms. *Mater. Today.: Proc.* **2022**, *62*, 4629–4634. <https://doi.org/10.1016/j.matpr.2022.03.080>. 671
672
31. Agilandeswari, L.; Prabukumar, M.; Radhesyam, V.; Phaneendra, K.L.N.B.; Farhan, A. Crop Classification for Agricultural Applications in Hyperspectral Remote Sensing Images. *Applied Sciences* **2022**, *12*, 1670. <https://doi.org/10.3390/app12031670>. 673
674
32. Tian, X.; Bai, Y.; Li, G.; Yang, X.; Huang, J.; Chen, Z. An Adaptive Feature Fusion Network with Superpixel Optimization for Crop Classification Using Sentinel-2 Imagery. *Remote Sensing* **2023**, *15*, 1990. <https://doi.org/10.3390/rs15081990>. 675
676
33. Rokach, L.; Maimon, O., Decision Trees. In *Data Mining and Knowledge Discovery Handbook*; Springer-Verlag; pp. 165–192. https://doi.org/10.1007/0-387-25465-x_9. 677
678
34. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. In Proceedings of the Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2016, Vol. 11, KDD '16, pp. 785–794. <https://doi.org/10.1145/2939672.2939785>. 679
680
681
35. Breiman, L. *Mach. Learn.* **2001**, *45*, 5–32. <https://doi.org/10.1023/a:1010933404324>. 682
36. Cortes, C. Support-Vector Networks. *Mach. Learn.* **1995**. 683
37. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. *Psychol. Rev.* **1958**, *65*, 386–408. <https://doi.org/10.1037/h0042519>. 684
685
38. Balestrieri, R.; Ibrahim, M.; Sobal, V.; Morcos, A.; Shekhar, S.; Goldstein, T.; Bordes, F.; Bardes, A.; Mialon, G.; Tian, Y.; et al. A Cookbook of Self-Supervised Learning, 2023. <https://doi.org/10.48550/ARXIV.2304.12210>. 686
687
39. *Machine Learning for Data Science Handbook: Data Mining and Knowledge Discovery Handbook*; Springer International Publishing, 2023. <https://doi.org/10.1007/978-3-031-24628-9>. 688
689
40. Lopez Pinaya, W.H.; Vieira, S.; Garcia-Dias, R.; Mechelli, A., Autoencoders. In *Machine Learning*; Elsevier, 2020; pp. 193–208. <https://doi.org/10.1016/b978-0-12-815739-8.00011-0>. 690
691
41. Lopez-Fandino, J.; Garea, A.S.; Heras, D.B.; Arguello, F. Stacked Autoencoders for Multiclass Change Detection in Hyperspectral Images. In Proceedings of the IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium. IEEE, 2018. <https://doi.org/10.1109/igarss.2018.8518338>. 692
693
694
42. Luppino, L.T.; Hansen, M.A.; Kampffmeyer, M.; Bianchi, F.M.; Moser, G.; Jenssen, R.; Anfinsen, S.N. Code-Aligned Autoencoders for Unsupervised Change Detection in Multimodal Remote Sensing Images. *IEEE Trans. Neural Netw. Learn. Syst.* **2024**, *35*, 60–72. <https://doi.org/10.1109/tnnls.2022.3172183>. 695
696
697
43. Kalinicheva, E.; Sublime, J.; Trocan, M., Change Detection in Satellite Images Using Reconstruction Errors of Joint Autoencoders. In *Artificial Neural Networks and Machine Learning – ICANN 2019: Image Processing*; Springer International Publishing, 2019; pp. 637–648. https://doi.org/10.1007/978-3-030-30508-6_50. 698
699
700
44. Lisaius, M.C.; Blake, A.; Keshav, S.; Atzberger, C. Using Barlow Twins to Create Representations From Cloud-Corrupted Remote Sensing Time Series. *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.* **2024**, *17*, 13162–13168. <https://doi.org/10.1109/jstars.2024.3426044>. 701
702
703
45. Windrim, L.; Ramakrishnan, R.; Melkumyan, A.; Murphy, R.J.; Chlingaryan, A. Unsupervised Feature-Learning for Hyperspectral Data with Autoencoders. *Remote Sensing* **2019**, *11*, 864. <https://doi.org/10.3390/rs11070864>. 704
705
46. Bégué, A.; Arvor, D.; Bellon, B.; Betbeder, J.; de Abelleyeira, D.; P. D. Ferraz, R.; Lebourgeois, V.; Lelong, C.; Simões, M.; R. Verón, S. Remote Sensing and Cropping Practices: A Review. *Remote Sensing* **2018**, *10*, 99. <https://doi.org/10.3390/rs10010099>. 706
707
47. Orynbailykyzy, A.; Gessner, U.; Conrad, C. Crop type classification using a combination of optical and radar remote sensing data: a review. *Int. J. Remote Sens.* **2019**, *40*, 6553–6595. <https://doi.org/10.1080/01431161.2019.1569791>. 708
709
48. Pierre Pott, L.; Jorge Carneiro Amado, T.; Augusto Schwalbert, R.; Mateus Corassa, G.; Antonio Ciampitti, I. Crop type classification in Southern Brazil: Integrating remote sensing, crop modeling and machine learning. *Comput. Electron. Agr.* **2022**, *201*, 107320. <https://doi.org/10.1016/j.compag.2022.107320>. 710
711
712
49. Moreno-Martínez, A.; Izquierdo-Verdiguier, E.; Maneta, M.P.; Camps-Valls, G.; Robinson, N.; Muñoz-Marí, J.; Sedano, F.; Clinton, N.; Running, S.W. Multispectral high resolution sensor fusion for smoothing and gap-filling in the cloud. *Remote Sens. Environ.* **2020**, *247*, 111901. <https://doi.org/10.1016/j.rse.2020.111901>. 713
714
715

50. Kandasamy, S.; Baret, F.; Verger, A.; Neveux, P.; Weiss, M. A comparison of methods for smoothing and gap filling time series of remote sensing observations – application to MODIS LAI products. *Biogeosciences* **2013**, *10*, 4055–4071. <https://doi.org/10.5194/bg-10-4055-2013>. 716
717
51. Tzelepi, M.; Nousi, P.; Passalis, N.; Tefas, A., Representation learning and retrieval. In *Deep Learning for Robot Perception and Cognition*; Elsevier, 2022; pp. 221–241. <https://doi.org/10.1016/b978-0-32-385787-1.00015-4>. 718
719
52. Reichstein, M.; Camps-Valls, G.; Stevens, B.; Jung, M.; Denzler, J.; Carvalhais, N.; Prabhat. Deep learning and process understanding for data-driven Earth system science. *Nature* **2019**, *566*, 195–204. <https://doi.org/10.1038/s41586-019-0912-1>. 720
721
53. Tschanne, M.; Djolonga, J.; Rubenstein, P.K.; Gelly, S.; Lucic, M. On Mutual Information Maximization for Representation Learning **2019**. [arXiv:cs.LG/1907.13625]. <https://doi.org/10.48550/ARXIV.1907.13625>. 722
723
54. Le-Khac, P.H.; Healy, G.; Smeaton, A.F. Contrastive Representation Learning: A Framework and Review. *IEEE Access* **2020**, *8*, 193907–193934. <https://doi.org/10.1109/access.2020.3031549>. 725
726
55. Aitchison, L.; Ganev, S. InfoNCE is variational inference in a recognition parameterised model, 2021. <https://doi.org/10.48550/ARXIV.2107.02495>. 727
728
56. Chen, T.; Kornblith, S.; Norouzi, M.; Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. In Proceedings of the Proceedings of the 37th International Conference on Machine Learning; III, H.D.; Singh, A., Eds. PMLR, 13–18 Jul 2020, Vol. 119, *Proceedings of Machine Learning Research*, pp. 1597–1607. 729
730
57. Bachman, P.; Hjelm, R.D.; Buchwalter, W. Learning Representations by Maximizing Mutual Information Across Views, 2019. <https://doi.org/10.48550/ARXIV.1906.00910>. 732
733
58. Dwibedi, D.; Aytar, Y.; Tompson, J.; Sermanet, P.; Zisserman, A. With a Little Help from My Friends: Nearest-Neighbor Contrastive Learning of Visual Representations, 2021. <https://doi.org/10.48550/ARXIV.2104.14548>. 734
735
59. Bardes, A.; Ponce, J.; LeCun, Y. VICReg: Variance-Invariance-Covariance Regularization for Self-Supervised Learning, 2021. <https://doi.org/10.48550/ARXIV.2105.04906>. 736
737
60. Zbontar, J.; Jing, L.; Misra, I.; LeCun, Y.; Deny, S. Barlow Twins: Self-Supervised Learning via Redundancy Reduction, 2021. <https://doi.org/10.48550/ARXIV.2103.03230>. 738
739
61. Coifman, R.R.; Lafon, S. Diffusion maps. *Appl. Comput. Harmon. A.* **2006**, *21*, 5–30. <https://doi.org/10.1016/j.acha.2006.04.006>. 740
62. Bank, D.; Koenigstein, N.; Giryes, R. Autoencoders, 2020. <https://doi.org/10.48550/ARXIV.2003.05991>. 741
63. Bank, D.; Koenigstein, N.; Giryes, R., Autoencoders. In *Machine Learning for Data Science Handbook*; Springer International Publishing, 2023; pp. 353–374. https://doi.org/10.1007/978-3-031-24628-9_16. 742
743
64. Zhang, L.; Qi, G.J.; Wang, L.; Luo, J. AET vs. AED: Unsupervised Representation Learning by Auto-Encoding Transformations rather than Data, 2019. <https://doi.org/10.48550/ARXIV.1901.04596>. 744
745
65. Baldi, P. Autoencoders, Unsupervised Learning, and Deep Architectures. In Proceedings of the Proceedings of ICML Workshop on Unsupervised and Transfer Learning; Guyon, I.; Dror, G.; Lemaire, V.; Taylor, G.; Silver, D., Eds., Bellevue, Washington, USA, 02 Jul 2012; Vol. 27, *Proceedings of Machine Learning Research*, pp. 37–49. 746
747
66. Dahlin, K.M.; Ponte, D.D.; Setlock, E.; Nagelkirk, R. Global patterns of drought deciduous phenology in semi-arid and savanna-type ecosystems. *Ecography* **2016**, *40*, 314–323. <https://doi.org/10.1111/ecog.02443>. 748
749
67. Foody, G.M. Status of land cover classification accuracy assessment. *Remote Sens. Environ.* **2002**, *80*, 185–201. [https://doi.org/10.1016/s0034-4257\(01\)00295-4](https://doi.org/10.1016/s0034-4257(01)00295-4). 750
751
68. 752
69. 753
70. 754
71. 755

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.