Computer Vision and Image Understanding
journal homepage: www.elsevier.com

Deep-STaR: Classification of image time series based on spatio-temporal representations

Mohamed Chelali[a,**], Camille Kurtz[a], Anne Puissant[b], Nicole Vincent[a]

[a] *Université de Paris, LIPADE, 75006, Paris, France*
[b] *Université de Strasbourg, LIVE, 67000, Strasbourg, France*

ABSTRACT

Image time series (ITS) represent complex 3D (2D+t in practice) data that are now daily produced in various domains, from medical imaging to remote sensing. They contain rich spatio-temporal information allowing the observation of the evolution of a sensed scene over time. In this work, we focus on the classification task of ITS, as often available in remote sensing tasks. An underlying problem here is to consider jointly the spatial and the temporal dimensions of the data. We present Deep-STaR, a method to learn such features from ITS data to proceed to their classification. Instead of reasoning in the original 2D+t space, we investigate novel 2D planar data representations, containing both temporal and spatial information. Such representations are a novel way to structure the ITS, compatible with deep learning architectures. They are used to feed a convolutional neural network to learn spatio-temporal features with 2D convolutions, leading ultimately to classification decision. To enhance the explainability of the results, we also propose a post-hoc attention mechanism, enabled by this new approach, providing a semantic map giving some insights for the taken decision. Deep-STaR is evaluated on a remote sensing application, for the classification of agricultural crops from satellite ITS. The results highlight the benefice of this method, compared to the literature, and its interest to make easier the interpretation of ITS to understand spatio-temporal phenomena.

1. Introduction

The multiplicity of sensors, coupled with the society appetites (*e.g.*, industrial, scientific, leisure) in image content, leads to the production of mass of visual data. They have to be processed, analyzed, understood automatically for indexing or classification purposes. In some cases, this visual data are $3D$ ($2D + t$ in practice) data when the sensors produce images of a scene at different times.

Such data sources are varied and many applications could benefit from them. In remote sensing, optical satellite sensors image certain regions every week. These data are used for environmental studies or land-cover mapping. For example, the Sentinel-2 Earth Observation satellite constellation provides image sequences over the same geographical area with high spatial, spectral and temporal resolutions around the globe (Drusch et al., 2012). In medicine, radiology imaging devices are used to follow each month the evolution of a pathology in a patient for longitudinal studies (Madhyastha et al., 2018). In biology, a camera fixed on a microscope can be employed to analyze the cell developments (Stuurman and Vale, 2016), etc.

The produced $2D + t$ data carry rich spatial and temporal information that must be taken into account to understand particular phenomena not being observable from a single image of the sequence (*e.g.*, vegetation seasonal development from satellite images, tumor remission in medicine) (Ren et al., 2009; Sumpter and Bulpitt, 2000; Weng et al., 2019).

Whether considering a stack of images or a video, we will denote these $2D + t$ data as Image Time Series (ITS) in the following. An ITS is basically a set of images of the same scene, ordered chronologically. It can be encoded as a data-cube, two spatial and one temporal dimensions. The acquisition of an ITS can be done with one or multiple sensors to obtain a larger data series with a high temporal frequency.

In this work, we consider a classification task where, given an ITS representing a scene or a particular object, a class la-

**Corresponding author
e-mail: mohamed.chelali@u-paris.fr (Mohamed Chelali),
camille.kurtz@u-paris.fr (Camille Kurtz),
anne.puissant@unistra.fr (Anne Puissant),
nicole.vincent@u-paris.fr (Nicole Vincent)

bel potentially linked to an evolution along time, has to be predicted. In addition, depending on the scene, moving objects or deformable content can be represented.

The analysis of an ITS generally requires the extraction, from image pixels, of visual features as discriminating as possible. In the literature, some approaches focus rather on the temporal aspect. They consider an ITS as a set of independent pixels characterized with their time series (*i.e.*, $1D$ temporal pixels) and classified individually. In a supervised classification scheme, this has the advantage of providing many learning examples to train a model. The spatial aspect of data is then totally ignored. Nevertheless, in various applications, this aspect is necessary to discriminate certain complex classes. The joint study of the spatial and temporal domains may allow a finer analysis and a better understanding of some phenomena which can characterize the studied objects of interest and their evolution. In this context, some approaches combine spatial and temporal features. Often, the two domains are processed independently and a fusion is operated for the decision. There are also approaches that directly take into account spatio-temporal features calculated from the data-cube, *e.g.*, convolutional features obtained from a $3D$ Deep Neural Network (DNN). Such features are then *natively* spatio-temporal but training such models is expensive.

The problem studied in this article is the extraction of spatio-temporal features from ITS and their involvement in a deep classification procedure. Our methodological contribution is twofold:

- we propose Deep-STaR, a method dedicated to ITS classification. We investigate novel planar representations of the $2D + t$ ITS data, involving both temporal and spatial information. The original $2D$ spatial dimension of the ITS is embedded in a $1D$ structure trying to preserve the pixels spatial configuration. Such $1D$ spatial structure is coupled with the $1D$ original temporal domain of the ITS in a $2D$ (planar) spatio-temporal representation, leading to a novel way to structure the ITS, making easier calculus and interpretation. This new representation is used to feed a Convolutional Neural Network (CNN) to learn spatio-temporal features, resulting ultimately to classification decisions;

- we investigate an attention mechanism, integrated in our system, providing a semantic map explaining the decision. The main originality is to embed the attention information in the original ITS dimensions. This constitutes a plus value regarding the state-of-the-art since attention was mainly studied in the spatial or temporal domains.

The remainder of this article is organized as follows. Section 2 introduces some related works. In Section 3, we present the Deep-STaR method: firstly, the proposed $2D$ spatio-temporal representations, secondly, the proposed attention mechanism. Sections 4 and 5 present an experimental study in remote sensing and a discussion of the results, coupled with a comparative study. Finally, conclusion and perspectives will be found in Section 6.

2. Related works for ITS analysis

Numerous approaches exist in the literature for ITS analysis. Depending on the task and the application field (*e.g.*, remote sensing, medical imaging, video analysis). We focus here on the features and the adopted point of view (*i.e.*, dimension). We distinguish three groups of approaches, presented hereinafter: (1) those treating ITS as a set of pixel time series, (2) those integrating spatial information in the analysis, and (3) those exploiting more directly spatio-temporal features.

2.1. ITS as pixel time series

Pioneer methods from the literature processed each image from the ITS, without considering the temporal information. Various colorimetric features were extracted at pixel level, image by image, in supervised machine learning-based procedures. In parallel, a lot of researches in the 90's, especially by the remote sensing community, addressed Satellite Image Time Series (SITS). The need to consider temporal information quickly appeared to study various types of changes and the evolution of the observed territories (Coppin et al., 2004).

Some methods consider only two images sensed at different dates. They study the transitions (*e.g.*, abrupt changes) between two observations, using for example image differencing (Bruzzone and Prieto, 2000) or rationing (Jensen, 1981). The change is then located by thresholding or classification. When several images are considered, the process is repeated for all succeeding couples of images.

Other methods consider all the images in the ITS. Some are based on a multi-date classification approach, such as radiometric trajectory analysis (Verbesselt et al., 2010). Another approach proceeds at pixel level by analyzing their evolution through time (Bagnall et al., 2017). Here, each pixel is viewed as a set of measurements ordered chronologically (*i.e.*, temporal pixel), and symbolic or statistical strategies have been proposed to analyze such patterns (Méger et al., 2019).

New representation spaces can be also considered for the analysis. The most well-known is "frequency-domain", such as Fourier transform or wavelet decomposition of the radiometric time series, involving auto-correlation and cross-correlation analysis (Andres et al., 1994). These methods require a regular temporal sampling. Besides, "hand-crafted" representation spaces defined by discriminative temporal features for classification task are proposed. The authors in (Chelali et al., 2019) propose temporal stability features from the image time series. The notion of temporal SIFT was proposed in (Bailly et al., 2015) to extract a compact set of features from temporal pixels. Finally, are available all the features processing $1D$ time-series such as when processing biomedical signals, financial data, industrial devices, etc.

The classification task is then applied in this new space ("hand-crafted" features from the temporal pixels) where classical approaches rely on similarity and on a training set. Similarity can be computed by different metrics such as an Euclidean distance or a Dynamic Time Wrapping (Petitjean et al., 2012a) measure. For example, nearest neighbor algorithm coupled with one of these metrics is used to assign the label of the most similar class.

Few years ago, DNN methods were employed for classification of temporal pixels. Two main families of DNN are generally considered: Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs). Initially, CNNs were conceived to analyze $2D$ images where convolutions deal only with spatial domain (Huang et al., 2018). They were adapted to time series. For example, the TempCNN architecture has been proposed for SITS classification (Pelletier et al., 2019) using $1D$ convolutions applied in the temporal domain. Classical CNNs, such as ResNet, have also been adapted for time series (Ismail Fawaz et al., 2019).

Another strategy is to encode the $1D$ time series into $2D$ representations, allowing then the use of a $2D$ CNN. For example, recurrence plots or the Gramian Angular Field (Wang and Oates, 2015) or the use of Short Time Fourier Transform to ensure the analysis of a signal varying through time (Nisar et al., 2016). Such strategies enable to benefit from powerful $2D$ CNN models, pre-trained on computer vision tasks (Russakovsky et al., 2015).

Concerning RNNs, the most popular is Long-Short Term Memory (LSTM) used successfully in (Ienco et al., 2017) for Earth observation applications. Researches show that DNN approaches for time series classification outperform the classical ones, such as Random Forest (Ismail Fawaz et al., 2019).

While these DNN-based approaches offer promising results, they consider (temporal) pixels independently, and generally neglect the image spatial dimensions. However, in certain applications, it is crucial to also consider the spatial aspect of the $2D + t$ data to discriminate accurately complex classes.

2.2. *Adding spatial information to characterize temporal pixels*

To address the limitations mentioned above, some researches integrated spatial information. The common idea is to characterize the $1D$ temporal pixels, not only with colorimetric or statistical information collected on each date, but also with spatial information collected in the pixels neighborhood.

Various strategies were proposed to integrate spatial information to temporal pixels. In (Petitjean et al., 2012b), an image segmentation of each image of the ITS enables to enrich a temporal pixel with information calculated in the regions it belongs to along the time axis. Different measurements are extracted from these regions: colorimetric, morphological, or even geometrical (Correa et al., 2020) or mean of temporal patterns as proposed in (Ravikumar and Devi, 2014).

Morphological attribute profiles and pyramids of pixel-based spatial features have also been considered to enrich temporal pixels with spatial information, generally for change detection tasks (Falco et al., 2013). Another approach relies on the Histogram of Oriented Gradient (HOG) considered to capture contextual information for pedestrian detection and tracking in videos (Barbu, 2014). Each image of the ITS can be also processed using $2D$ CNNs to extract convolutionnal features that are stacked and passed to a classifier to operate a global classification of the data-cube (Tran et al., 2018).

In the context of agricultural crop-fields classification from SITS, the authors of (Sainte Fare Garnot et al., 2020) recently proposed an hybrid strategy relying on DNN. First, sub-sets of pixels are considered randomly from the crop-fields images. A spatial encoding operator, inspired from $3D$ point cloud processing, enables a network to learn first order statistical descriptors of the data spectral distribution. They are considered as spatial features in the model but neighborhood information is ignored. Such features are then combined with temporal features extracted using a neural architecture based on self-attention, to ultimately produce a classification result.

These approaches make it possible to combine spatial information with temporal information, but such features are not natively spatio-temporal.

2.3. *Spatio-temporal features*

Spatio-temporal features denote features that are computed by involving simultaneously the spatial and the temporal domains of an ITS.

Depending on the considered tasks, numerous "hand-crafted" spatio-temporal features in video analysis were investigated such as $3D$ SIFT (Scovanner et al., 2007), texture features, or based on motion with optical-flow strategies, etc. Most of research about such features were done in video analysis for action recognition, detection of video copies, which mostly made use of such features.

DNN-based approaches have been considered for learning spatio-temporal features (Atluri et al., 2018). In (Chandra et al., 2018), authors consider the spatial and the temporal domains separately, training two models, one on each domain, with an aggregation at the decision level. Authors of (Di Mauro et al., 2017) propose to use $1D$ CNN in the temporal domain. Then some spatial features are computed, such as the mean of pixels within a radius at each time, and are combined with the temporal features. Since the approach considers the pixel coordinates as spatial features, final results are visually rather smooth. Other strategies use a network with both $1D$ and $3D$ convolutions alternating (Stoian et al., 2019). Note that deep spatio-temporal features are classically learned in a supervised fashion but auto-encoders architectures can also be considered (Goroshin et al., 2015).

Eventually, $3D$ convolutions can be directly employed to deal with the spatial and temporal domains simultaneously (Tran et al., 2015). A $3D$ convolutional auto-encoder is proposed in (Kalinicheva et al., 2020) for segmentation purpose. In (Tran et al., 2018), spatio-temporal convolutions for video analysis are studied. Their conclusions demonstrate the accuracy advantages of $3D$ CNNs over $2D$ CNNs applied to individual frames of the video for action recognition purpose. The authors of (Feichtenhofer et al., 2017) propose a two-stream networks, one trained on RGB video and the second on motion video (optical flow). These networks are linked with residual connections in order to learn the interaction between appearance and motion. This strategy is well adapted for scenes with deformations and motions, but less competitive on (non-deformable) static scenes. We find also the DuPLO method (Interdonato et al., 2019) that applies 2D convolutions on an image that is built as a multi-band input considering as many bands as the length of the time series. If such features are natively spatio-temporal, training such models remains expensive. Furthermore, models

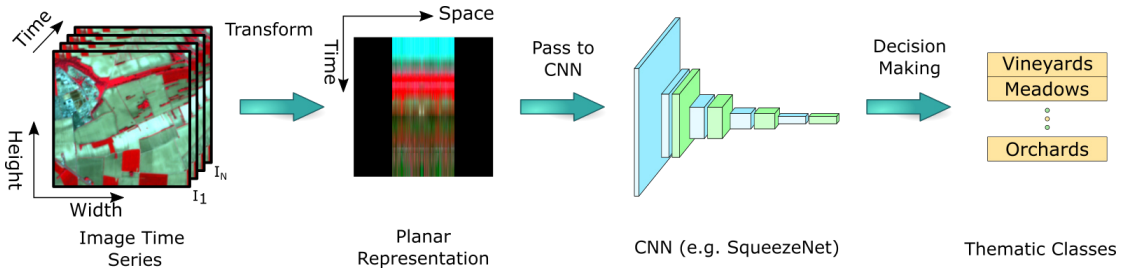


Fig. 1. Deep-STaR method relying on spatio-temporal planar representations.

with millions of parameters such as $3D$ CNNs can lead to solutions difficult to interpret, in particular with $2D + t$ data.

2.4. Understanding CNN decision to improve ITS analysis

Generally, CNN models are trained and are more active on some parts of images or signals, according to the input. CNN are known as black boxes, due to their lack of interpretability. Indeed, millions of parameters are learned, in particular with large architectures dedicated to $2D + t$ or $3D$ data analysis.

In this context, different attention mechanisms strategies were introduced. One of them is trainable attention mechanism helping the models to focus on some parts of the input during the training (Jetley et al., 2018). However, this requires the modification of the model architecture. This can be tricky when using classical pre-trained architectures. There are also post-hoc attention mechanism that introduces the notion of Class Activation Map (CAM) (Zhou et al., 2016), highlighting the most important image parts for the model to take the decision. Note that CAM is only employable with models having a Global Average Pooling (GAP) layer before the softmax classifier (Zhou et al., 2016). CAM are optimized by using a guided back-propagation and are compatible with CNN models ending with a fully connected layer, see *e.g.*, GradCAM and GradCAM++(Chattopadhyay et al., 2018). The notion of CAM is also used when considering temporal convolutions of time series (Fawaz et al., 2019).

When dealing with ITS, attention-based approaches have also been proposed (Xu et al., 2020). Very often, the attention is captured independently in the spatial and temporal domains and then combined. Thus, it does not enable to improve directly the understanding of spatio-temporal phenomena in the $2D + t$ data, leading to the decision of the network.

2.5. Motivations

In this article, we focus on spatio-temporal features learning using DNN for ITS classification. As discussed in the previous sub-sections, a lot of studies consider only temporal aspect. Considering an ITS as a bag of independent pixel time series, the studies benefit from large sets of samples for supervised learning procedures. On the opposite, complex $3D$ CNN methods learn (native) spatio-temporal features but such approaches need a huge database to fix all the parameters and they can lead to solutions difficult to interpret.

In our case, we want to benefit from the advantages of both approaches: a large number of training entities for learning and both spatial and temporal information from the original $2D + t$ data-cube. To this end, we propose an ITS classification method based on novel spatio-temporal representations embedding both temporal and spatial information in a $2D$ image structure. Such representations allow a classical $2D$ CNN to learn $2D$ filter weights extracting (native) spatio-temporal features for the ITS classification.

Based on our previous work (Chelali et al., 2020), where we proposed only a global $2D$ representation of the data, we introduce here a new local strategy. This is an original methodology that makes it possible to increase the number of representations associated with the ITS and to control it, deeply improving the model learning step. We also propose a novel post-hoc attention mechanism, providing a semantic map, whose originality is to embed attention information in the original ITS $2D + t$ space to better understand the CNN decision.

3. Deep-STaR: ITS analysis from spatio-temporal representations

This section presents the methodological foundation of the proposed Deep-STaR method for $2D + t$ image time series to predict a semantic (class) label from an input ITS. Figure 1 illustrates the workflow of Deep-STaR.

The ITS can be either a (rectangular) patch representing a complete scene (see left of Figure 1), or only a region of interest (ROI), a connected set of pixels in the image domain (see Figure 7). We assume that all pixels of the patch/ROI share the same label. In the following, we will use the term “image” and ITS to refer to such input data.

The core of Deep-STaR is based on the concept of spatio-temporal planar representations. Rather than the original ITS $2D + t$ space, we investigate novel $2D$ planar representations containing both temporal and spatial information. The original $2D$ spatial dimension of the ITS is transformed to a $1D$ space using different strategies (which can operate at local or global level) trying to preserve the spatial configuration by partially maintaining the neighborhood of the pixels. This novel $1D$ spatial dimension is combined with the original $1D$ temporal dimension of the ITS, leading to a $2D$ spatio-temporal representation. Such representation can be considered as a novel way to structure the ITS, making easier its manipulation and also its interpretation. This new kind of representation feeds a classical $2D$ Convolutional Neural Network (CNN) which learns spatio-temporal features with $2D$ convolutions, leading ultimately to classification decision.

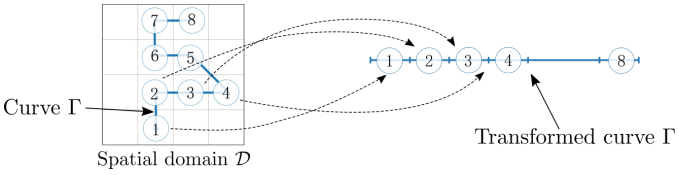
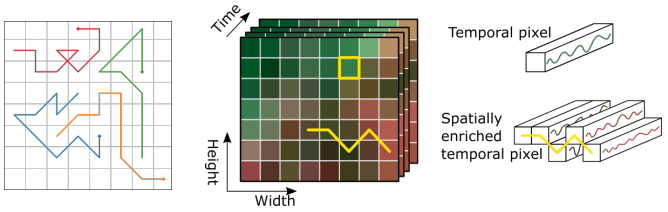


Fig. 2. $2D$ **image transformation to a** $1D$ **pixel array: (left) A curve** Γ **in the spatial image domain** $\mathcal{D}$**; (right) Representation of** Γ **in a** $1D$ **structure of length** $L = 8$**, the pixels are indexed according to their curvilinear abscissa.**



(a) RW segments (b) A spatially enriched temporal pixel

Fig. 3. Random Walk segments and enrichment of pixel time series with spatial information: (a) RW segments (one color per segment) built in the spatial domain $\mathcal{D}$ **of the image (the starting points of the segments are materialized with dots); (b) An example of a temporal pixel (*i.e.*, pixel time series) versus a spatially enriched temporal pixel in ITS.**

After providing notations (Section 3.1), Section 3.2 deals with the construction of the planar representations. Then, Section 3.3 describes the different entities that are used to transform the $2D + t$ data-cube into $2D$ structures. Sections 3.4 and 3.5 present the CNN architecture and the decision making process for the ITS labelling. Finally, Section 3.6 presents another contribution relying on a post-hoc attention mechanism to better understand the decision taken thanks to the proposed system.

3.1. Notations

In the following, an ITS is represented as a tuple $(I_n)_{n\in[\![1,N]\!]}$ of images, n is the acquisition date of image I_n. An image I is a function associating an integer couple $(x, y) \in \mathcal{D}$ (*i.e.*, a pixel) with its intensity (represented by one or more radiometric values). The (common) spatial domain of the images is $\mathcal{D} = [0, H-1] \times [0, W-1]$ where W and H represent respectively the width and the height of the images. We assume here that all images in the series are registered, and a pixel covers the same scene surface at all dates.

3.2. Planar representation construction

Before dealing with the $2D+t$ data, we start by defining how to transform $2D$ images into $1D$ arrays of pixels.

Given an image I, the spatial information is figured by neighboring pixels. A pixel in a $2D$ space or in a region has (in general) 4 or 8 nearest neighbors directly connected according to the considered topology. This makes a spatial context. In order to simplify the complexity of the $2D$ structure, the idea is to limit the pixel neighbors in a $1D$ structure with length L, bringing partial local spatial information. In order to keep statistically significant information, we select curves where pixels have 2 neighbors except curve extremities, ensuring an isotropic exploration of the space. Then, a $1D$ structure is defined by indexing pixels according to their curvilinear abscissa within the curve. Doing so, in each curve, spatial information is decreased as a pixel will have only 2 neighbors among the 8 possible ones. Note that our objective is to preserve as much as possible the spatial information.

The way we consider the ITS is based on the way we are looking at an image, *i.e.*, the entities we are focusing on. Here, we focus on curves drawn in the $2D$ image, it can be a more of less short curve or a curve that fills the whole image domain, making possible both local or global data investigations.

Let us consider a curve Γ in the spatial image domain $\mathcal{D}$. We note the curve pixels P_j; they are indexed by their curvilinear abscissa initiated at one of the curve extremity. Then, Γ is represented by the tuple $(P_1, \ldots, P_L)$, where L is the length of the $1D$ structure. At this stage, the curve Γ embedded in the $2D$ space is such, naturally represented in a $1D$ structure, *i.e.*, a string of pixels. Figure 2 illustrates this strategy.

The next step is to handle $2D + t$ data. The curve Γ defined in the image support $\mathcal{D}$ has occurrences in all the N images of the ITS. Then, the N tuples $P = \{\left(P_1^i, \ldots, P_L^i\right)\}_{i=1}^N$ (one tuple per date) are stacked as rows in a $2D$ array, interpreted as a new image; such image is called a Spatio-Temporal Representation (STR). The STR size is $N \times L$. N, the height refers to the temporal aspect (the number of images in the ITS), L the width refers to the spatial aspect (the curve pixel index). Finally, the new $2D$ image represents a planar representation of the $2D + t$ ITS. Note that if the pixels P_j in the original ITS were characterized by multiple radiometric values (*i.e.*, multivalued images), then the STR is a multivalued $2D$ image.

We will present different strategies to draw curves in the image domain, at a local or a global level, associating different sets of tuples with an ITS.

3.3. Curves carrying significant spatial information

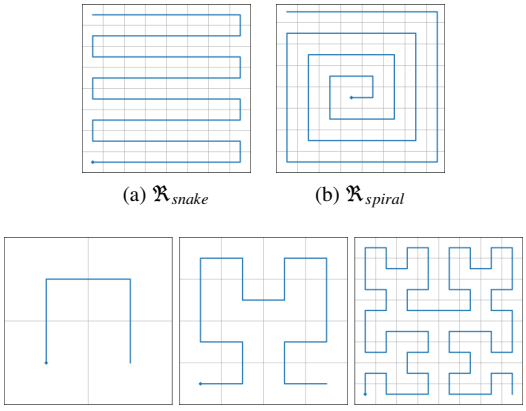
The main purpose is to introduce spatial information when building a planar spatio-temporal representation. Different curves in the image plane are possible, either at local level (see Section 3.3.1), leading to a Multi-Segment Spatio Temporal Representation (MS-STR), or at global level (see Section 3.3.2), leading to a Global Spatio Temporal Representation (G-STR).

3.3.1. Local point of view: Random Walk strategy

As supervised learning procedures often require a large dataset for training, a local (pixel) point of view can be adopted in order to enrich pixel time series with spatial information. Such local strategy has the advantage of making numerous different entities from an ITS.

Technically, the idea is to extract several ($1D$) curve segments from the image plane, capturing local spatial information through a limited number of pixels belonging to the ROI and building ITS planar representations as explained previously. The way segments are built is now introduced.

Random Walk segments (RW): The goal is to pass from single pixels to segments. Here, we propose to build curve segments by a Random Walk (RW) process. RW is a mathematical model to build a path from random iterative steps with Markovian properties. The RW segment with length L is noted



(a) $\mathfrak{R}_{snake}$ (b) $\mathfrak{R}_{spiral}$

(c) $\mathfrak{R}_{Hilbert}$ (three first orders)

Fig. 4. Space-filling curves transforming a $2D$ image in a $1D$ pixel string.

$RW(L)$. The first point of the curve is initialized randomly. 8 directions are possible for choosing next point except if a pixel is next to the region border. Figure 3 presents some examples of $RW(L)$ with different L values and how pixel time series can be enriched with spatial information.

Depending on the starting points of the curves and thanks to the randomness of the process, N_{seg} representations can be built, sharing the same label. They model the ITS, leading to a Multi-Segment Spatio Temporal Representation (MS-STR). N_{seg} is then a parameter enabling to control the number of different representations of a ROI.

3.3.2. *Global point of view: space filling curves*

Unlike previously, here we propose a global view of the image domain $\mathcal{D}$ leading to a Global Spatio Temporal Representation (G-STR). For this, a curve filling the whole plane is considered. In the literature, many strategies exist to index image pixels; among them, (Nguyen et al., 2012) proposes to use space filling curves passing through all the pixels of a square $2D$ plane only once. The most important property of space filling curves is locality preserving. Two adjacent pixels in the curve are neighboring pixels in the plane.

Pixels indexed initially by (x, y) can be indexed by only one integer value j. Let a function $\mathfrak{R}$ be dedicated to this transformation:

$$\begin{array}{rccc} \mathfrak{R}: & \mathcal{D} & \rightarrow & [0,(W-1)\times(H-1)] \\ & (x,y) & \mapsto & j=\mathfrak{R}(x,y) \end{array} \quad (1)$$

Different space-filling curves exist. Each one has its proper strategy to keep statistically representative neighbors without bias. Here, we choose and compare experimentally different curves described below:

- **Snake curve ($\mathfrak{R}_{snake}$):** this curve scans the plane lines, as a snake (see Figure 4(a)). To preserve the spatial relationship, lines are linked, the heads of even lines are linked with ends of odd ones, and *vice versa*;
- **Spiral curve ($\mathfrak{R}_{spiral}$):** this curve is based on Archimedean spiral that fills the $2D$ square plane, as illustrated in Figure 4(b). The square center is the curve first point. Then, the curve revolves around;
- **Hilbert curve ($\mathfrak{R}_{Hilbert}$):** this curve is a fractal space-filling curve (Butz, 1971). The construction of the curve is based on a recurrent process applied on a square domain. The domain is divided into four equal squares. The four small squares are linked in such a way that "two parts with a common edge have two consecutive indexes". This rule is applied recursively on squares with power of 2 as width. Figure 4(c) illustrates the three first orders of the process.

Once the $2D$ spatial domain is transformed to $1D$ structures (local or global strategy), temporal information can be added as described in Section 3.2, leading to the final planar STR images (MS-STR or G-STR). Next step is to train a model exploiting these representations in a deep architecture.

3.4. *CNN architecture for STR labelling*

Generally, CNNs are involved in end-to-end methods to analyze the visual aspect of images. Their architectures differ. But the main process is to pass an input through the network. It starts by a succession of layers involving convolution operations followed by an activation function (*e.g.*, sigmoïde, ReLU) to select only the high order features from the input. Another layer, max-pooling, is an operation to reduce the quantity of the inputs for next layers. Finally, in classification systems we find fully connected layers, that have the same principle as a multi-layer perceptron, followed by a softmax function to get the predicted probabilities of the classes.

Our method is based on a $2D$ CNN to learn and extract spatio-temporal features with $2D$ convolutions. Our choice comes to SqueezeNet (Iandola et al., 2016) but it can be changed with any other $2D$ CNN. SqueezeNet is a rather small network leading to the same accuracy level as AlexNet when evaluated on the ImageNet dataset. A Fire layer is introduced in the model. It applies 1×1 convolution followed by two expand layers, one applies 1×1 convolution and the second 3×3 convolution. The network is the concatenation of the previous expand layers. Inputs of the SqueezeNet are 224×224 images.

The STR (local or global) are used as inputs to train the CNN. Each STR has the label of the ITS it belongs to. The learning can use the data available but can also benefit from transfer learning strategies. Here, as we are using image data, the computed weights on the ImageNet dataset classification task can be fine-tuned with our STR images.

3.5. *Decision making*

The decision making process is relatively similar for both local or global approaches. In the local approach, many curves can be extracted from the ITS domain (*i.e.*, a MS-STR contains N_{seg} STR). In the global one, only specific $2D$ representations are associated with the ITS. In any case, the CNN provides as output a class probability vector for each STR input. For the local strategy, for each ITS to be classified, N_{seg} probability vectors are obtained.

In both cases, the final decision for the ITS is taken by averaging the returned probabilities given by the CNN classifier

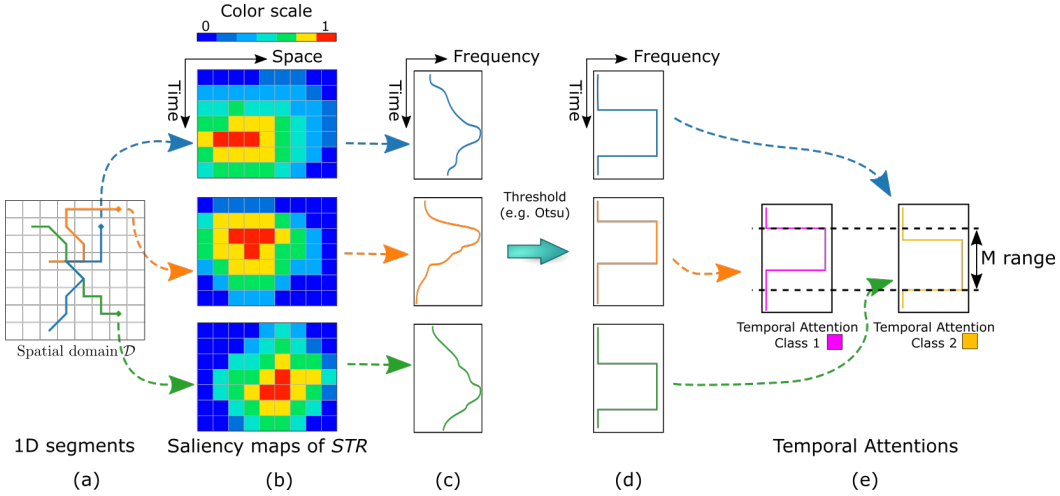


Fig. 5. Temporal attention in MS-STR approach: (a) RW segments (a dot materializes starting point); (b) Saliency maps of N_p STR (colored attention values); (c) Temporal attention profiles; (d) Thresholded profiles; (e) Mask M for global temporal attention.

applied to the available STR modeling the ITS. State-of-the-art methods focusing on the temporal domain generally provide decision at temporal pixel level, *i.e.*, one decision per pixel, and then average the final results to get a unique decision per ROI. We apply the same strategy to get a single decision per ITS.

With the classification, we propose a visual understanding of the taken decision using an attention mechanism to analyze more deeply the learned spatio-temporal features, and to justify the taken decision.

3.6. Attention mechanism

The decision depends on the most active reception fields. Thanks to the learned weights, a CNN computes features. For example, we can find a pedestrian or a car in an image but both can be found and, as a by product result, a segmentation is built by the CNN. In this context, the use of a Class Activation Map (CAM) strategy leads to a saliency map providing a visualization of which image parts contribute the most to each label.

In our experimental study, we classified a ROI using a CNN but without more detailed analysis. The identification of the agricultural classes depends on the vegetation cycles (because of the seasons) and also from the farmers' management of the parcels by adding fertilizer, mowing some parts of parcel or only using a part from the parcel. Adding to the classification task, the proposed planar representation is used to identify the influence of the temporal and the spatial aspects that are respectively figured along the height and the width of the image.

We provide hereinafter a method for studying temporal and spatial attentions from the (STR) planar spatio-temporal representations.

3.6.1. Temporal attention

Using temporal information, we assume the temporal evolution is meaningful to discriminate between the different classes. Some questions may occur: What is the most discriminative temporal moment? Is it necessary to study the ITS on the whole available period? Finally, can the results be improved by considering a specific period?

We assume that time stamp is identical for all the samples used in the training, and that all spatial areas may contribute in the same way to the decision. The STR input to the CNN has in the abscissa information from a spatial point of view and in the ordinates from a temporal point of view. Then, to focus on temporal attention, we will consider the vertical axis.

For each STR image, the GradCAM++ (Chattopadhyay et al., 2018) strategy leads to a saliency map S^c for a class $c \in [1, C]$. The saliency map can also be viewed as a $N \times L$ attention image (see Figure 5(a,b)). Let us note $S^c_{i,j}$ the attention at date $i \in [1, N]$ (row) and pixel $j \in [1, L]$ (column) in S^c. By accumulating values on the rows in S^c, we obtain a temporal attention vector with size N, noted X. Each element of the temporal attention vector is defined as:

$$X_i(STR) = \sum_{j=1}^{L} S^{c(STR)}_{i,j} \tag{2}$$

where X_i is the i^{th} coordinate of vector X and $X(STR)$ is the temporal attention vector associated with STR image. We remind that S^c is obtained for one class c. In our case, S is computed only once for predicted class of the STR image. Figure 5(c) illustrates the obtained temporal attention vectors for a STR. From the analysis of all the vectors associated with the learning elements, we can select the most discriminant temporal range for the CNN. This is done by applying a thresholding A to embed the X vectors in $\{0, 1\}^N$ giving AX hard vectors. The high values in X are those with 1 in AX and inversely. The Otsu method is used for A, illustrated in Figure 5(d).

We can then build a mask M that represents the global temporal range enabling the discrimination between classes. It is based on the AX vectors. Within each class, we compute the component wise product of the AX^k. This highlights the most significant times for the classification of this class (see Figure 5(e)). To classify the C classes, instead of studying the time series on the whole period, the period can be shorten, considering only the times where at least one $(AX^c)_i$ equals 1 as:

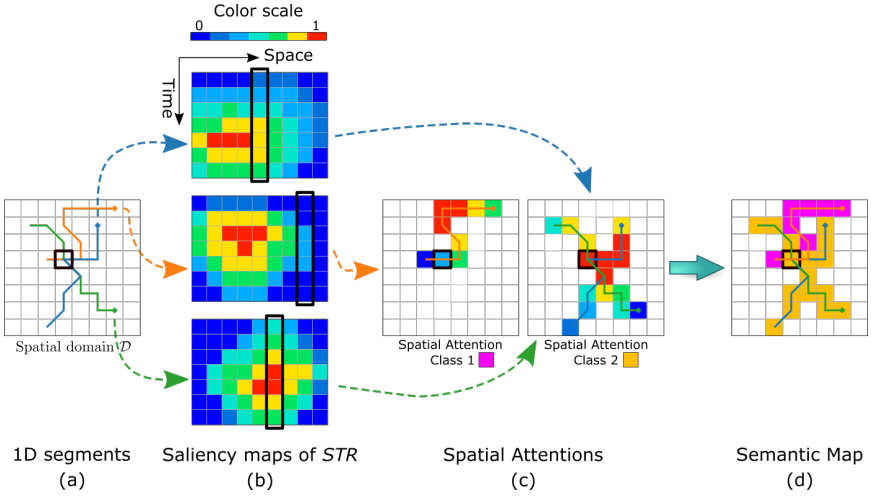


Fig. 6. Spatial attention in MS-STR approach: (a) RW segments (a dot materializes starting point); (b) Saliency maps of the N_p STR (attention values are colored); (c) retro-projection of the attention values in the image domain $\mathcal{D}$; (d) resulting semantic map.

$$M = (M_i)_i = \begin{cases} 1 & \text{if } \exists k \in [1, C] \quad \prod_{STR/c(STR)=k} AX_i^{c(STR)} = 1 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

3.6.2. Spatial attention

The saliency map S^c highlights regions on the spatio-temporal representation image domain, that contribute more to the decision taking. We present now a method enabling to highlight the interesting spatial information in the original $2D$ image domain $\mathcal{D}$ of the ITS.

In a STR, the horizontal axis carries the spatial information. In a similar way as for temporal attention, for each pixel we consider its column in the saliency map S^c, but instead of averaging, we consider the time where the pixel is the most attractive for class c and define vector Y as:

$$Y_j^c = Y_{(x,y)}^c = \max_{i=1}^{N} S_{i,j}^c \tag{4}$$

where Y_j^c is the attention associated with pixel of index j in the STR image and of coordinates $(x, y) \in \mathcal{D}$ in the original image domain.

Of course, the retro-projection process cannot be the same when global or local methodologies are applied for building the STR. In a global approach (G-STR), it is possible to build an image where a pixel of coordinates (x, y) has value equal to $Y_{(x,y)}^c$. The process is more complex with a local approach (MS-STR). The region is represented by a set of N_{seg} STR (P^k, $k \in [1, N_{seg}]$), all with the same length. Spatially, each pixel of coordinates $(x, y) \in \mathcal{D}$ belongs to a set of Γ curves, associated with P^k, $k \in K$. Each P^k corresponds to a STR image with c_k as predicted label. From the saliency map, pixel (x, y) has an attention value $Y_{(x,y)}^{c_k}$, associated with P^k, noted $Y_{(x,y)}^{c_k,P^k}$. Finally, the spatial attention of pixel (x, y) relative to class c is defined:

$$Y_{(x,y)}^c = \max\left(Y_{(x,y)}^{c_k,P^k} \; k \in K \text{ and } c_k = c, \;\; 0\right) \tag{5}$$

This is illustrated in Figure 6(a) with a pixel belonging to 3 curves, two labeled c_1 and one labeled c_2. The 3 saliency maps are presented in Figure 6(b). In each case, the considered pixel is associated with one column in the saliency map and we consider the spatial attention of the initial pixel is the highest value in the column (formula 4), one value per representation. When considering the class label c, the highest value among the STR decisions giving c is considered as showed in Figure 6(c).

To get a semantic map, we propose to build a domain map where is indicated the pixel label giving the highest attention among possible classes:

$$V_{(x,y)} = \arg\max_{c \in C} \max_{k \in K/c=c_k} Y_{(x,y)}^{c_k,P^k} \tag{6}$$

Such a domain / semantic map is illustrated in Figure 6(d) with 2 classes.

4. Experimental study in remote sensing

Deep-STaR is experimented on a remote sensing application. Recently, new Earth Observation satellite constellations sense masses of satellite image time series (SITS). The Sentinel-2 provides image sequences over a geographical area with high spatial, spectral and temporal resolutions. Such $2D + t$ imaging data are useful for agricultural and environmental policy makers, since they enable for example the control of agricultural crop-fields at large-scale to check the annual farmers declarations. Our objective is to classify four thematic classes of SITS: (1) meadows, (2) vineyards, (3) traditional orchards and (4) intensive orchards. The automatic identification of these classes is complex since these agricultural crop-fields are subject to many agricultural practices depending on the seasons and the territory management policies. The thematic study of these classes is motivated by a multidisciplinary research project (funded by the French National Research Agency), including geomatics and computer scientists[1]. Previous studies

[1]TIMES project – *High-performance processing techniques for mapping and monitoring environmental changes from massive, heteroge-*

Table 1. Data summary: (second col.) Initial number of polygons per class; (third col.) Average number of pixels per polygon; (five last col.) Number of spatio-temporal segments.

Classes	# poly.	$\overline{area}$		# MS-STR				# G-STR
		mean	std	$RW_{10\%}$	$RW_{20\%}$	$RW_{50\%}$	$RW_{70\%}$	$\mathfrak{R}_*$
Meadows	1 045	250	338	26 110	51 688	128 424	179 914	1 757
Vineyards	562	50	47	3 060	5 821	14 137	19 853	577
Trad. orchards	136	154	305	2 146	4 222	10 474	14 672	189
Int. orchards	191	129	115	2 564	5 027	12 414	17 408	226
Total	1 934	–	–	33 880	66 758	165 449	231 847	2 749

also highlighted that orchards are ambiguous classes and tend to be confused in most classification methods (Stoian et al., 2019). To differentiate these classes, spatio-temporal features carry rich information to discriminate the agricultural practices, tree organizations or mowing, etc.

4.1. Materials

We studied SITS acquired with the Sentinel-2 satellite, containing $N = 50$ images acquired in 2017 over the same area in East of France, precisely in 32ULU and 32ULV tiles. Figure 7 displays samples from four SITS and the temporal distribution of the N images. The images were pre-processed by the French Theia program to correct and orthorectify them to be radiometrically comparable. Masks of cloud, shadow and saturation are given for each image.

In our experiments, we only consider three spectral bands: near-infrared (Nir), red (R) and green (G), with a 10 meters spatial resolution. The blue band is too much sensitive to atmospheric effects (Pelletier et al., 2019), and does not allow to discriminate between agricultural classes. Also, we kept only three bands in order to be similar to other possible computer vision contexts, involving classical RGB images.

4.2. Reference data

The used reference data are extracted from the French Land Parcel Identification System records[2], providing polygons with their semantic labels. We kept only the polygons corresponding to the four studied classes: meadows, vineyards, traditional orchards and intensive orchards. Second column of Table 1 displays the number of polygons. A photo-interpretation is applied to correct, if needed, the crop-field delimitations. The satellite images are cropped to build the ITS. Figure 7 illustrates some examples of agricultural crop-fields.

4.3. Data preparation

The aim is to classify agricultural crop-field, modeled as ROIs. The STR images from these ROIs with different local and global strategies are constructed (see Figure 8). They are adapted to fit the input size of the CNN, both in the temporal and the spatial dimensions, as explained hereinafter.

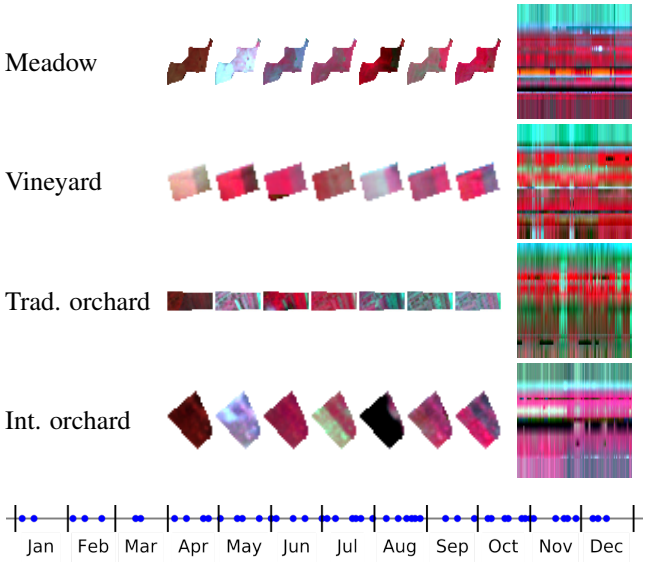


Fig. 7. Temporal evolution of four SITS, according (left) to their initial $2D + t$ representation from a SITS and (right) to our $2D$ global spatio-temporal representation. The last line shows the distribution of the images from the SITS (2017).

Temporal dimension (vertical axis). To fill the full size of the CNN input image, we apply a linear interpolation on the temporal dimension to generate 224 dates from the SITS assuming monotonic and linear evolution between two dates. By doing so, we: (1) fill the missing values, or masked values by clouds; (2) re-sample the temporal series to obtain 224 dates.

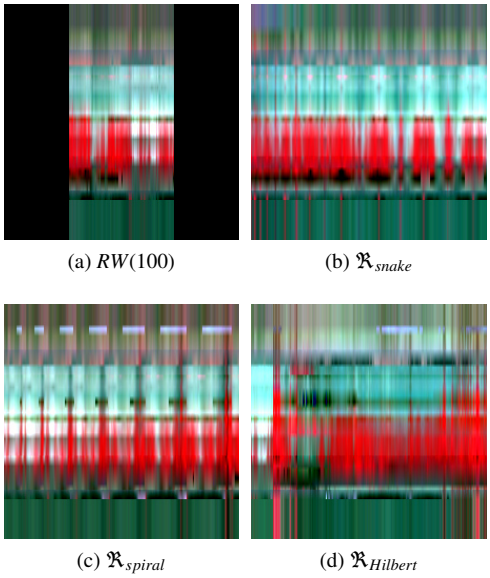
Spatial dimension (horizontal axis). According to the considered strategies, the spatial dimension is handled differently and detailed below.

Local strategy (MS-STR): to get a large training dataset, N_{seg} segments are extracted from each ROI, initialized differently in the RW process, leading to MS-STR. N_{seg} is fixed as respectively 10%, 20%, 50% and 70% of the number of pixels in the considered polygon. For instance if a ROI is composed of 100 pixels, with parameter 70%, we build 70 different planar representations to model the ROI. Table 1 (columns 4–7) shows the number of STR built. We also indicate the mean area of ROIs with their standard deviation, noted $\overline{area}$.

By varying the length L of the segments, we study the importance of the spatial information enriching the pixel time series in the STR. The studied lengths are 10, 50 and 100. Finally,

neous and high frequency data times series, see https://anr.fr/Projet-ANR-17-CE23-0015

[2] http://professionnels.ign.fr/rpg



(a) $RW(100)$ (b) $\mathfrak{R}_{snake}$

(c) $\mathfrak{R}_{spiral}$ (d) $\mathfrak{R}_{Hilbert}$

Fig. 8. Four STR built with (MS-STR, G-STR) strategies for a meadow ITS: (a) Local approach with a Random Walk strategy ($RW(100)$). 124 black columns are added to fill the 224×224 image size; (b, c, d) Global approach with three different space-filling curves.

we center the obtained planar representations on the horizontal axis of the CNN input image and we set zero values for the other pixels of the input STR image.

Global strategy (G-STR): depending on the number of pixels of the ROI, two strategies enable to have 224 column images: (1) for the ROIs less than 224 pixels, we repeat the sequence until the 224 values in X axis are filled; (2) for those with more than 224 pixels, we split the sequence into different images with 224 values. In Table 1 (last column) the number of G-STR generated is higher than the initial number of polygons per class.

According to the number x of STR associated with a ROI, the same decision making is employed, see Section 3.5, where N_{seg} is replaced by x.

4.4. Data normalization

Data normalization plays a crucial role in pattern recognition task. This pre-processing re-scales the data in such a way comparisons are possible between the image samples. The most often used technique is the z-normalization. It proceeds by subtracting from each pixel value the mean of the image pixel values and then by dividing by the standard deviation of the set of pixel values. In our case, the z-normalization is not adapted since, due to the acquisition process, there are some outliers that would disturb the process.

Another traditional normalization is based on the maximum and the minimum values in the dataset. Then, from every scalar data is subtracted the minimum value (min) and the result is divided by ($max - min$), where max is the maximum value of the scalar data. We then opted for such normalization technique which is more robust with some further constraint: we limited the values with 2% (or 98%) percentile, as proposed in (Pelletier et al., 2019).

4.5. Data augmentation for global approaches

From Table 1, it can be noticed that the datasets for the local approaches are relatively large (due to the multiple STR representations per ROI built from the RW process) but in case of the global approach only few samples are available. As we are considering an end-to-end deep learning procedure to learn spatio-temporal features and to classify the data, this low number of data can interfere with the learning of a good model.

In the case of the global approaches, we then investigated data augmentation (DA) techniques to get more training data to use in the learning step. In order to get more annotated data, multiple DA techniques can be considered. Traditional DA methods are based on basic image transformations. The most common are affine transformations such as flipping, rotation and translation. Also, some are non-affine ones such as resizing, cropping or adding noise. More recent methods are based on deep generative-adversarial neural networks to produce additional synthetic data (Shorten and Khoshgoftaar, 2019). In our case, such GAN methods cannot be considered to generate more data since we need a huge annotated database which is not available in the context of this thematic study. In this study, we only use classical affine transformations applied on the initial spatial domain $\mathcal{D}$ in order to conserve the initial resolution of the input and the size of polygons. Then the generated STRs will be different from each others. We apply some rotations with the specific angles: 45°, 90°, 135° and 180°.

4.6. Learning and validation protocol

During the experiment, the dataset is split into three data subsets at parcel (or ROI) level. The subsets represent respectively the training, validation and test sets. Their sizes are respectively 60%, 20% and 20% of the total number of labeled data available, respecting the proportions of the classes. We use a 5-fold cross validation technique to evaluate the model performance. Then, we report the average overall accuracy (OA) with the standard-deviation (STD).

Adam optimizer is used with a learning rate of 10^{-6} and default values of the other parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$) with a batch size of 128. To handle the unbalanced classes, we use a weighted cross entropy loss function to avoid a potential overfitting towards the majority classes. We use an early stop with a patience number of 20. The experiments are done on a server with a Nvidia GPU model Tesla T4. We used the PyTorch implementation of SqueezeNet[3]. The network is initialized with weights obtained from the ImageNet dataset, then we fine-tune with our images.

5. Results and discussion

We discuss here the classification results on the remote sensing application, with the local MS-STR and global G-STR approaches and present some comparisons with selected competitive methods from the state-of-the-art. We finally provide visual results obtained with the attention mechanisms.

[3] https://pytorch.org/hub/pytorch_vision_squeezenet/

Table 2. Global classification results (overall accuracy – OA and stand. deviation – STD).

	Rep.	# Rep.	OA	STD
Deep-STaR	*Rand*	10%	88.87	1.56
		20%	89.02	1.38
		50%	**90.66**	**0.85**
		70%	90.00	1.13
	Local approaches (MS-STR)			
	$RW(10)$	10%	90.51	0.48
		20%	91.48	0.75
		50%	92.56	0.95
		70%	**93.07**	**1.02**
	$RW(50)$	10%	91.07	2.53
		20%	93.80	1.57
		50%	94.06	1.44
		70%	**94.80**	**1.57**
	$RW(100)$	10%	92.50	1.05
		20%	93.20	0.65
		50%	94.21	1.19
		70%	**94.64**	**0.80**
	Global approaches (G-STR)			
	$\mathfrak{R}_{snake}$	w/o DA	79.94	2.06
	$\mathfrak{R}_{spiral}$	w/o DA	77.23	1.42
	$\mathfrak{R}_{Hilbert}$	w/o DA	**81.69**	**1.88**
	$\mathfrak{R}_{snake}$	with DA	91.43	1.58
	$\mathfrak{R}_{spiral}$	with DA	89.43	1.61
	$\mathfrak{R}_{Hilbert}$	with DA	**91.69**	**0.91**
State-of-the-art methods	TempCNN (Pelletier et al., 2019)	–	92.98	0.89
	baML (Di Mauro et al., 2017)	–	91.25	0.53
	LSTM (Ienco et al., 2017)	–	83.48	2.29
	ConvLSTM (Rußwurm and Körner, 2018)	–	74.66	1.56
	$3D$-SqueezeNet (Köpüklü et al., 2019)	–	**85.33**	**1.19**

Table 3. Inference time in seconds for our best models vs. state-of-the-art methods (sorted in the increasing order).

Rep.	Average time
G-STR $\mathfrak{R}_{Hilbert}$	2.72
baML Di Mauro et al. (2017)	11.91
TempCNN (Pelletier et al., 2019)	13.50
MS-STR $RW(50)$	22.57
ConvLSTM (Rußwurm and Körner, 2018)	23.16
$3D$-SqueezeNet (Köpüklü et al., 2019)	26.50
LSTM (Ienco et al., 2017)	26.96

5.1. STR visualization

Before presenting quantitative results, we illustrate in Figure 8 some STR built with the different strategies of Section 3.2. The G-STR images differ. With $\mathfrak{R}_{snake}$, regular patterns appear horizontally while with $\mathfrak{R}_{spiral}$, the patterns become larger because the curve starts in the center of the image and goes farther with larger amplitude. As expected, $\mathfrak{R}_{Hilbert}$ provides a result smoother than the others because this curve preserves better the locality of pixels. For the local approach, *RW* provides a smooth result like the $\mathfrak{R}_{Hilbert}$.

5.2. Global classification results

In order to better illustrate the interest of our methodology, we add a naive baseline involving no spatial information. STR are constructed as sequence of random pixel time series, noted *Rand* with $L = 10$. With such a baseline, the resulting STR can be considered as bag of pixel time series.

As a preliminary experiment, we conducted an ablation study by training the model with two strategies. The first one is to train the model on a restricted temporal range, by keeping one date out of two, leading to time series with lengths of 112 dates. The second strategy is to consider all the 224 dates. Here we consider only the MS-STR with $RW(50)_{70\%}$. From our observations, the better results are those obtained with the 224 dates, exceeding with 4% the results with restricted temporal time-stamps. In the following, we then kept this strategy.

Table 2 reports the obtained global classification scores with our Deep-STaR models. As expected, within the local strategy, the random strategy (*Rand*) is the weakest. For the *RW*, whatever the length of the segments is, the scores increase with the segment number. The highest result is obtained when $L = 50$, this length appears a good compromise between short curves giving not enough information on a pixel region, and long curves that can cover non homogeneous region.

The global G-STR approaches appear to be less efficient than the local MS-STR approaches. This is partially due to the small size of the available learning set, emphasized by the improvement thanks to data augmentation. In addition, the G-STR are all in a 4-connexity topology. This could explain the lowest results observed. In the MS-STR approach, the randomness of the orientation of the RW enables to have an isotropic view of the local region. However, in the best condition of the global approach, the overall accuracy is greater than with *Rand*.

We also perform the experiments by training the CNN model from scratch. From our observations, the obtained scores are lower than when a fine-tuning strategy is considered. The difference is about 4% for the MS-STR and about 10% for G-STR. Such a difference for the G-STR may be due to the lower quantity of data when considering global representations. Also, the training is much quicker when fine-tuning is considered.

As a comparative study, we compared our scores on the data to five state-of-the-art methods:

- TempCNN (Pelletier et al., 2019) dedicated to pixel time series classification, where $1D$ convolutions are applied in the temporal domain;
- baML, a deep convolutional method that works separately on the spatial and on the temporal domain proposed in

Table 4. Obtained per class scores (precision – P, Recall – R and F1-Measure – F1).

Classes		Meadows	Vineyards	Trad. orchards	Int. orchards	Average
Deep-STaR						
MS-STR - *RW(50)*$_{70\%}$	*P*	96.08	99.47	78.81	88.02	90.59
	R	95.80	98.58	91.42	81.01	91.70
	*F*1	**95.94**	99.02	**84.53**	**84.16**	**90.91**
G-STR - $\mathfrak{R}_{Hilbert}$	*P*	94.56	96.70	71.40	79.71	85.59
	R	92.66	96.84	85.18	75.89	87.64
	*F*1	93.59	96.75	77.16	77.72	86.30
State-of-the-art methods						
TempCNN (Pelletier et al., 2019)	*P*	90.98	99.82	89.47	85.42	91.42
	R	99.29	99.82	48.57	73.04	80.18
	*F*1	94.94	**99.82**	61.78	78.68	83.80
baML (Di Mauro et al., 2017)	*P*	93.51	96.75	67.17	74.12	82.88
	R	97.14	99.12	54.07	62.56	78.22
	*F*1	95.28	97.92	58.86	67.72	79.94
LSTM (Ienco et al., 2017)	*P*	95.58	96.07	34.58	48.80	68.75
	R	82.76	98.77	47.40	67.69	74.15
	*F*1	88.70	97.40	39.91	56.55	70.64
ConvLSTM (Rußwurm and Körner, 2018)	*P*	84.95	84.94	12.07	26.88	52.21
	R	80.66	93.33	11.85	31.28	54.28
	*F*1	82.75	88.94	11.96	28.91	53.14
3*D*-SqueezeNet (Köpüklü et al., 2019)	*P*	91.41	94.79	39.84	56.21	70.56
	R	88.09	98.77	29.63	96.74	78.30
	*F*1	89.58	96.73	33.96	62.36	70.65

(Di Mauro et al., 2017);

- a custom RNN composed of 3 layers of LSTM with 256 hidden states and a fully connected layer, inspired from (Ienco et al., 2017);

- a convolutional RNN composed of 2 layers of ConvLSTM with 64 hidden states and a convolution layer followed by a relu and batch normalization, inspired from (Rußwurm and Körner, 2018). The classifier is fully convolutional followed by a global average pooling;

- 3*D*-SqueezeNet (Köpüklü et al., 2019), is a 3*D* extension of SqueezeNet network, proposed for human action recognition within video. In its implementation (Köpüklü et al., 2019), the input is a series of 16 frames. Here, we select 16 frames using a regular step. The model is pre-trained on the *Jester* dataset.

For comparison purpose, we use the same learning and validation protocol (Section 4.6). The bottom part of Table 2 reports the obtained scores. Overall, the Deep-STaR approaches provide the best classification scores. 3*D*-SqueezeNet (Köpüklü et al., 2019) gives disappointing results. Such approach suffers from the lack of material in the learning phase. This highlights, in our context, the benefit of considering a classical 2*D* CNN model for classifying image sequences combined with our spatio-temporal representations. Best score of state-of-the-art methods is obtained with TempCNN (Pelletier et al., 2019). The parameters of the method have been optimized on our data. We used a filter size of 11 related to the series length (Pelletier et al., 2019). In the original paper, the model is proposed with various depths of the network but it does not affect much the scores and we indicate here only the best one.

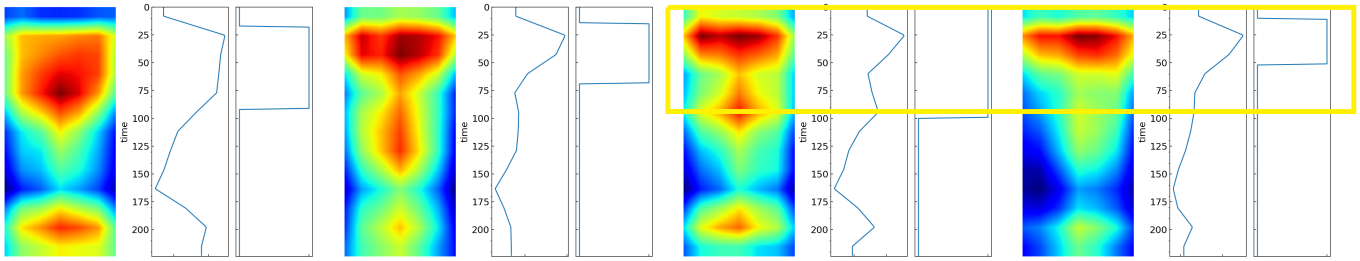
This highlights the interest of holding as much as possible spatial relationships between pixel time series. Locally, *RW* preserves partially neighbours, depending on the curve length *L*. Globally, pixels proximity is involved too.

Table 3 indicates the inference time for state-of-the-art methods as well as our two best models. The model based on the G-STR $\mathfrak{R}_{Hilbert}$ representations has the lowest inference time as the number of the STR is limited (only few per ROI, see Section 4.3 and Table 1). In the second position, we find the baML method (Di Mauro et al., 2017) and TempCNN (Pelletier et al., 2019). baML is faster than TempCNN since their model does not have high number of parameters compared to TempCNN. Our proposal, the local representations with $RW(50)_{70\%}$, comes in the third position as the number of images is higher and 2*D* images are considered. Finally, ConvLSTM (Rußwurm and Körner, 2018), 3*D*-SqueezeNet (Köpüklü et al., 2019) and LSTM (Ienco et al., 2017) are in the last positions due to the complexity and the memory management of the methods.

5.3. Per-class classification results

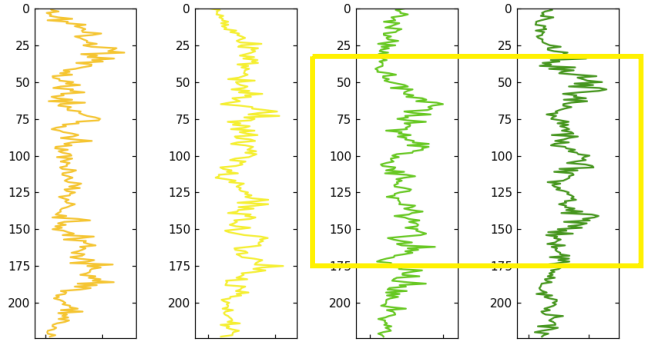
To go further and since the dataset is not well-balanced, we report in Table 4 the per-class results with our best methods and state-of-the-art ones.

$RW(50)_{70\%}$ provides the best per-class scores. Deep-STaR is particularly powerful to discriminate categories of orchards (not the most represented classes). Such agricultural crop-fields are characterized with spatial arrangements of trees and peculiar (temporal) agricultural practices.



(a) Vineyards (b) Meadows (c) Traditional orchards (d) Intensive orchards

Fig. 9. Illustration of the temporal attention computed with Deep-STaR on our thematic application. The mean of attention maps of the four classes are provided with their associated temporal attention profiles and their binarizations (see Section 3.6.1). Yellow rectangles represent temporal ranges of interest considered in a 2-class study (traditional vs. intensive orchards).



(a) Vineyards (b) Meadows (c) Traditional orchards (d) Intensive orchards

Fig. 10. Illustration of the temporal attention for the four classes obtained with TempCNN (Pelletier et al., 2019). Yellow rectangles represent temporal ranges of interest considered in a 2-class study (traditional vs. intensive orchards).

Finally the confusion matrices (not provided here) show that too much traditional orchards are classified as intensive ones by Deep-STaR. All approaches tend to confuse meadows and orchards, in particular TempCNN (Pelletier et al., 2019).

To analyze the CNN behavior, we used the attention mechanisms to figure which parts of the STR are more used and also to understand some errors.

5.4. Attention maps results

Figures 9 illustrates the obtained temporal attention maps on the four agricultural classes with Deep-STaR. The mean of attention maps of the four classes are provided with their associated temporal attention profiles and their binarizations.

To better evaluate the interest of this approach, we focus here on a 2-class study, involving traditional and intensive orchards. From the associated temporal attention profiles, we select the most discriminative period (see Section 3.6.1). Such period corresponds to a temporal range between time 1 and 100 (*i.e.*, January to mid-June). Applying the model respectively on these new temporal ranges (see yellow rectangles in Figure 9), the results have been improved, highlighting the interest of the study of attention maps in discarding non significant period of the year according for the problem. More precisely, the decrease in errors is about 7%. For comparison purpose, we conducted a comparable study, by generating temporal attention maps on the four agricultural classes with TempCNN (Pelletier et al., 2019), see Figure 10. Yellow rectangles represent again the new temporal range (between 25 and 195 (*i.e.*, February to November)) considered in the 2-class study. By re-training the model on these restricted intervals, the decrease in errors is about 5% with TempCNN (Pelletier et al., 2019).

Besides accuracy improvement, a conclusion of this study is that temporal attention enables to understand, from the SITS, that Spring season was more significant than the rest of the year to make the discrimination.

Figure 11 illustrates our study about spatial semantic maps on some examples of landscapes involving meadows. Meadows are large areas where agricultural practices can vary and evolve over time (*e.g.*, reforestation, new crops) according to the needs of landowners. They constitute very heterogeneous objects of interest where a single class label may not be very consistent and this heterogeneity could lead the classifier to errors. For visualization purpose, the spatial attention maps are compared with very high spatial resolution images from Google Earth (a, b, c). This enables to see what is happening in the spatial neighborhood of the ROIs. The maps (d, e, f) are obtained with Deep-STaR and our best MS-STR model with $RW(10)$. Meadows (a, b) are well classified in the meadow class, whereas meadow (c) is misclassified as traditional orchard. Moreover, in the spatial semantic maps, almost all pixels of meadows (a, b) are labeled as meadows (yellow color) in (d) and (e). First, we can notice on image (c) the supposed meadow comprises many isolated trees. This explains why, on the spatial semantic map, a majority of pixels of the ROI are labeled as traditional orchard (light green color). Also the right part of this ROI contains vineyard and it is well labeled thanks to the spatial attention mechanism.

We also compare the obtained semantic maps with those obtained with TempCNN (Pelletier et al., 2019) (g, h ,i) and 3D-SqueezeNet (Köpüklü et al., 2019) (j, k, l). TempCNN classifies well all meadows (a, b, c). The obtained semantic maps with TempCNN are almost homogeneous with some pixels classified as orchards and vineyards. In order to get a segmentation

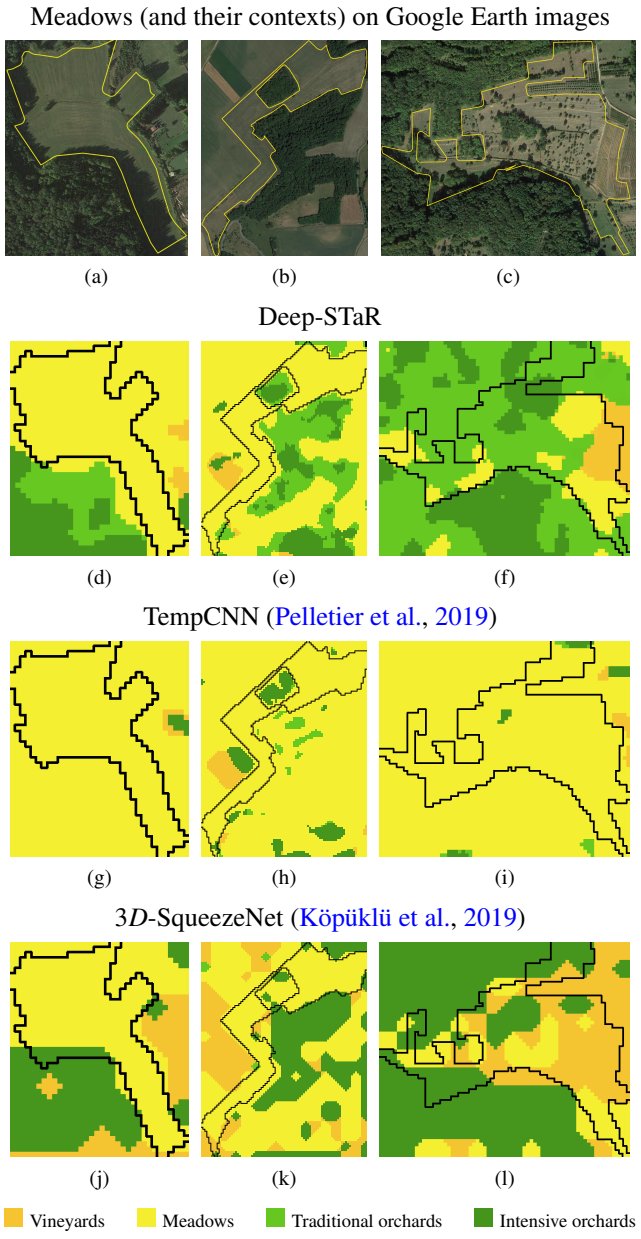


Fig. 11. Illustration of the semantic maps obtained with Deep-STaR and state-of-the-art methods: (a, b, c) Bounding boxes of three meadows (and their contexts) represented on a very high spatial resolution image from Google Earth (meadow borders are in yellow); (d, e, f) Spatial semantic maps based on the spatial attention with our best MS-STR model obtained with $RW(10)$; (g, h, i) Spatial semantic maps obtained with TempCNN (Pelletier et al., 2019); (j, k, l) Spatial semantic maps obtained with 3D-SqueezeNet (Köpüklü et al., 2019). Meadow borders are in black.

with 3D-SqueezeNet, we start by splitting the spatial domain into small patches. Then, patches are classified and we affect a specific color according to the predicted label. 3D-SqueezeNet classifies well meadows (a, b) and fails on the meadow (c). In the maps obtained with 3D-SqueezeNet, tree areas are always labelled as an intensive orchards. We remark that in the meadow (b), the common point of all segmentation with the different methods is the vineyards area that is located in the same position. Beyond this detail, the concordance between the field observations (from the Google Earth satellite image) and the results obtained with Deep-STaR compared to the state of the art confirms again the interest of our method. This experiment highlights how semantic maps can explain the spatial context of the conclusion, enabling for example to set an alert flag on some heterogeneous polygons.

6. Conclusion

In this work, we have proposed the Deep-STaR method designed for image time series classification. Thanks to a remodeling of the image time series into a planar spatio-temporal representation, spatial relationship of pixels is partially preserved, without losing the temporal information and native spatio-temporal features are learned while training a classical 2D CNN. The use of a 2D CNN allows to benefit of pre-learned weights, extracted from ImageNet and fine-tuned with specific data. Two strategies are proposed to analyze the preserved spatial configurations, local and global strategies, MS-STR and G-STR, depending on the objective.

Deep-STaR, experimented on a remote sensing application dedicated to agricultural crop-field classification, showed scores better than the state-of-the-art method scores, highlighting the interest of the proposed method. The classification task becomes simpler and also benefit from models that are trained on larger datasets. By integrating an original attention mechanism, a more discriminant temporal range for different thematic classes can be adapted. Also, generating spatial semantic maps helps to have a fine interpretation of the taken decision.

Our methodology can be applied on other applications, for example in biomedical image analysis where cell modifications are studied along time. Video indexing is another perspective. Besides we think important to deeper analyze the filters leading to spatio-temporal features. This could enable to better understand the properties of the ROI that is studied.

Acknowledgements

The French ANR supported this work under Grant ANR-17-CE23-0015.

References

Andres, L., Salas, W., Skole, D., 1994. Fourier analysis of multi-temporal AVHRR data applied to a land cover classification. Int J Remote Sens 15, 1115–1121.

Atluri, G., Karpatne, A., Kumar, V., 2018. Spatio-temporal data mining: A survey of problems and methods. ACM Comput Surv 51, 154–169.

Bagnall, A., Lines, J., Bostrom, A., Large, J., Keogh, E., 2017. The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. Data Min Knowl Discov 31, 606–660.

Bailly, A., Malinowski, S., et al., R.T., 2015. Dense bag-of-temporal-sift-words for time series classification, in: AALTD@ECML/PKDD, Procs., pp. 17–30.

Barbu, T., 2014. Pedestrian detection and tracking using temporal differencing and hog features. Comput Electric Enginee 40, 1072 – 1079.

Bruzzone, L., Prieto, D., 2000. Automatic analysis of the difference image for unsupervised change detection. IEEE Trans Geosci Remote Sens 38, 1171–1182.

Butz, A., 1971. Alternative algorithm for Hilbert's space-filling curve. IEEE Trans on Computers 20, 424–426.

Chandra, S., Couprie, C., Kokkinos, I., 2018. Deep spatio-temporal random fields for efficient video segmentation, in: CVPR, Procs., pp. 8915–8924.

Chattopadhyay, A., Sarkar, A., Howlader, P., Balasubramanian, V.N., 2018. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks, in: WACV, Procs., pp. 839–847.

Chelali, M., Kurtz, C., Puissant, A., Vincent, N., 2019. Urban land cover analysis from satellite image time series based on temporal stability, in: JURSE, Procs., pp. 1–4.

Chelali, M., Kurtz, C., Puissant, A., Vincent, N., 2020. Image time series classification based on a planar spatio-temporal data representation, in: VISAPP, Procs., pp. 276–283.

Coppin, P., Jonckheere, I., Nackaerts, K., Muys, B., Lambin, E., 2004. Digital change detection methods in ecosystem monitoring: A review. Int J Remote Sens , 1565–1596.

Correa, Y.T.S., Bovolo, F., Bruzzone, L., Fernández-Prieto, D., 2020. A method for the analysis of small crop fields in sentinel-2 dense time series. IEEE Trans Geosci Remote Sens 58, 2150–2164.

Di Mauro, N., Vergari, A., Basile, T.M.A., Ventola, F.G., Esposito, F., 2017. End-to-end learning of deep spatio-temporal representations for satellite image time series classification, in: DC@PKDD/ECML, Procs., pp. 1–8.

Drusch, M., Bello], U.D., et al., S.C., 2012. Sentinel-2: Esa's optical high-resolution mission for gmes operational services. Remot Sens of Environm 120, 25 – 36.

Falco, N., Mura, M.D., Bovolo, F., Benediktsson, J.A., Bruzzone, L., 2013. Change detection in VHR images based on morphological attribute profiles. IEEE Geosci Remote Sens Lett 10, 636–640.

Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P., 2019. Deep learning for time series classification: a review. Data Min Knowl Discov 33, 917–963.

Feichtenhofer, C., Pinz, A., Wildes, R.P., 2017. Spatiotemporal multiplier networks for video action recognition, in: CVPR, Procs., pp. 7445–7454.

Goroshin, R., Bruna, J., Tompson, J., Eigen, D., LeCun, Y., 2015. Unsupervised feature learning from temporal data, in: ICLR, Procs.

Huang, B., Lu, K., Audebert, N., Khalel, A., Tarabalka, Y., Malof, J., Boulch, A., 2018. Large-scale semantic classification: Outcome of the first year of inria aerial image labeling benchmark, in: IGARSS, Procs., pp. 6947–6950.

Iandola, F., Moskewicz, M., Ashraf, K.e.a., 2016. Squeezenet: AlexNet-level accuracy with 50x fewer parameters and <1MB model size. arXiv abs/1602.07360.

Ienco, D., Gaetano, R., Dupaquier, C., Maurel, P., 2017. Land cover classification via multitemporal spatial data by deep recurrent neural networks. IEEE Geosci Remote Sens Lett 14, 1685–1689.

Interdonato, R., Ienco, D., Gaetano, R., Ose, K., 2019. DuPLO: A DUal view Point deep Learning architecture for time series classificatiOn. ISPRS Journal of Photogrammetry and Remote Sensing 149, 91–104.

Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller, P., 2019. Deep learning for time series classification: A review. Data Min Knowl Discov 33, 917–963.

Jensen, J.R., 1981. Urban change detection mapping using Landsat digital data. Cart and Geog Inf Sci 8, 127–147.

Jetley, S., Lord, N.A., Lee, N., Torr, P.H.S., 2018. Learn to pay attention, in: ICLR, Procs.

Kalinicheva, E., Sublime, J., Trocan, M., 2020. Unsupervised satellite image time series clustering using object-based approaches and 3d convolutional autoencoder. Remote Sens 12.

Köpüklü, O., Kose, N., Gunduz, A., Rigoll, G., 2019. Resource efficient 3d convolutional neural networks, in: ICCV workshops, Procs., pp. 1910–1919.

Madhyastha, T., Peverill, M., Koh, N., McCabe, C., Flournoy, J., Mills, K., King, K., Pfeifer, J., McLaughlin, K.A., 2018. Current methods and limitations for longitudinal fmri analysis across development. Develop Cogn Neuro 33, 118–128.

Méger, N., Rigotti, C., Pothier, C., et al., T.N., 2019. Ranking evolution maps for satellite image time series exploration: application to crustal deformation and environmental monitoring. Data Min Knowl Discov 33, 131–167.

Nguyen, G., Franco, P., Mullot, R., Ogier, J., 2012. Mapping high dimensional features onto Hilbert curve: Applying to fast image retrieval, in: ICPR, Procs., pp. 425–428.

Nisar, S., Khan, O.U., Tariq, M., 2016. An efficient adaptive window size selection method for improving spectrogram visualization. Comput Intell Neurosci 2016, 1–13.

Pelletier, C., Webb, G., Petitjean, F., 2019. Temporal convolutional neural network for the classification of satellite image time series. Remote Sens 11, 523–534.

Petitjean, F., Inglada, J., Gançarski, P., 2012a. Satellite image time series analysis under time warping. IEEE Trans Geosci Remote Sens 50, 3081–3095.

Petitjean, F., Kurtz, C., Passat, N., Gançarski, P., 2012b. Spatio-temporal reasoning for the classification of satellite image time series. Patt Rec Letters 33, 1805–1815.

Ravikumar, P., Devi, V.S., 2014. Weighted feature-based classification of time series data, in: CIDM, Procs., pp. 222–228.

Ren, W., Singh, S., Singh, M., Zhu, Y., 2009. State-of-the-art on spatio-temporal information-based video retrieval. Pattern Rec 42, 267 – 282.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L., 2015. ImageNet Large Scale Visual Recognition Challenge. Int J Comput Vis 115, 211–252.

Rußwurm, M., Körner, M., 2018. Convolutional lstms for cloud-robust segmentation of remote sensing imagery, in: Spatiotemp@NIPS, Procs.

Sainte Fare Garnot, V., Landrieu, L., Giordano, S., Chehata, N., 2020. Satellite image time series classification with pixel-set encoders and temporal self-attention, in: CVPR, Procs.

Scovanner, P., Ali, S., Shah, M., 2007. A 3-dimensional sift descriptor and its application to action recognition, in: MM, Procs., pp. 357–360.

Shorten, C., Khoshgoftaar, T.M., 2019. Journ big data. JBD 6, 60–72.

Stoian, A., Poulain, V., Inglada, J., Poughon, V., Derksen, D., 2019. Land cover maps production with high resolution satellite image time series and convolutional neural networks: Adaptations and limits for operational systems. Remote Sens 11, 1986.

Stuurman, N., Vale, R.D., 2016. Impact of new camera technologies on discoveries in cell biology. Biol Bull 231, 5–13.

Sumpter, N., Bulpitt, A., 2000. Learning spatio-temporal patterns for predicting object behaviour. Image and Vision Computing 18, 697 – 704.

Tran, D., Bourdev, L., Fergus, R., Torresani, L., Paluri, M., 2015. Learning spatio-temporal features with 3D convolutional networks, in: ICCV, Procs., pp. 4489–4497.

Tran, D., Wang, H., Torresani, L., Ray, J., et al., Y.L., 2018. A closer look at spatiotemporal convolutions for action recognition, in: CVPR, Procs., pp. 6450–6459.

Verbesselt, J., Hyndman, R., Newnham, G., Culvenor, D., 2010. Detecting trend and seasonal changes in satellite image time series. Remote Sens Environ 114, 106–115.

Wang, Z., Oates, T., 2015. Imaging time-series to improve classification and imputation, in: IJCAI, Procs., pp. 3939–3945.

Weng, J., Weng, C., Yuan, J., Liu, Z., 2019. Discriminative spatio-temporal pattern discovery for 3d action recognition. IEEE Trans Circuits Syst Video Techn 29, 1077–1089.

Xu, Q., Xiao, Y., Wang, D., Luo, B., 2020. Multiscale octave 3d cnn with channel and spatial attention for hyperspectral image classification. Remote Sens 12.

Zhou, B., Khosla, A., Lapedriza, À., Oliva, A., Torralba, A., 2016. Learning deep features for discriminative localization, in: CVPR, Procs., pp. 2921–2929.