

# Temporal Vegetation Modelling using Long Short-Term Memory Networks for Crop Identification from Medium-Resolution Multi-Spectral Satellite Images

Marc Rußwurm and Marco Körner  
Remote Sensing Technology  
Technical University of Munich, Germany  
{marc.russwurm, marco.koerner}@tum.de

## Abstract

Land-cover classification (LCC) is one of the central problems in earth observation and was extensively investigated over recent decades. In many cases, existing approaches concentrate on single-time and multi- or hyper-spectral reflectance measurements observed by spaceborne and airborne sensors. However, land-cover classes, such as crops, change their reflective characteristics over time, thus complicating a classification at one particular observation time. Opposed to that, these characteristics change in a systematic and predictive manner, which should be utilized in a multi-temporal approach.

We employ long short-term memory (LSTM) networks to extract temporal characteristics from a sequence of SENTINEL 2A observations. We compared the performance of LSTM networks with other architectures and a support vector machine (SVM) baseline and show the effectiveness of dynamic temporal feature extraction. For our experiments, a large study area together with rich ground truth annotations provided by public authorities was used for training and evaluation. Our rather straightforward LSTM variant achieved state-of-the-art classification performance, thus opening promising potential for further research.

## 1. Introduction

In remote sensing and earth observation, *land cover classification (LCC)* is one of the key challenges due to its wide-ranging applicability. For instance, the *European Union* calculates the amount of agricultural subsidies based on reports of farmers about their cultivated crop, which could possibly be controlled via earth observation methods. Besides, more and more work has been carried out in detection of illegal crops based on air- and spaceborne imagery during the last decades, but most applications still rely on tedious visual inspection by experts.

In general, LCC systems from the domain of earth obser-

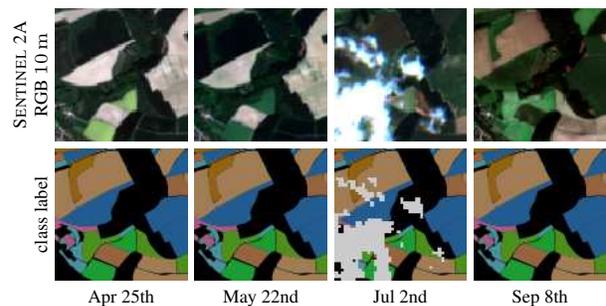


Figure 1. Sequence of observations along the growth season 2016. Observed fields change in a systematic and predictive manner based on crop phenology, which can be utilized for classification.

vation solely examine multi-spectral sensor data at individual ground positions or their surrounding regions acquired at a specific point in time and excluding cloud-covered observations. The spectral reflectances of crops change along the growth season due to their individual crops phenology, machining agriculture, and environmental conditions (*cf.* Figure 1). For these reasons, spaceborne sensors with high temporal resolution (*i.e.*, one day), such as MODIS, have been used in large-scale land cover classification [1, 2] for many years. Although their ground resolution of 250 m at nadir is not detailed enough for small scale LCC, these sensors have been used widely in regional and global monitoring tasks.

Thus, we believe that—especially in such settings—additional temporal modelling is called for and may perform superior to mono-temporal modelling schemes [3]. While this idea used to be hard to realize due to rather limited access to eligible data, SENTINEL2-A/B and LANDSAT-7/8 satellites now deliver medium-resolution multi-spectral remote sensing data at high temporal resolution, *i.e.*, with a revisit time of five days.

On the downside, due to these increased data stocks, intelligent methods for handling large amounts of data efficiently are in high demand. In addition, manual model design of nat-

ural vegetation processes is tedious or even impossible due to complex relations of internal biochemical processes, the inherent relations between environmental variables, and the unclear crop behaviour. Hence, considering the success of recent *deep learning* techniques and challenges of per-plot classification, we propose to employ *end-to-end* learning principles to model the crop vegetation cycle. By doing so, we can turn the big data drawback into an asset.

Thus, as main the contributions of this paper, we

- i) present a concept for processing temporal information, as provided by SENTINEL 2 and LANDSAT-7/8 satellites, in Section 3,
- ii) evaluate different data partitioning schemes in training and evaluation datasets in the context of spatial correlation in the data in Section 4.2.1, and
- iii) examine the influence of temporal links between observations by monitoring classification accuracies of temporal and non-temporal models in Section 4.2.2.

## 2. Related Work

While vegetation analysis with continuous monitoring over the growth season dates back many decades [4], only recently spaceborne sensors provide sufficient ground sampling distance and temporal resolution for single-plot field classification. Thus, classical approaches for land-cover classification usually do not take temporal information into consideration. These systems are most commonly composed of sequential building blocks—*e.g.*, data preprocessing, feature extraction, classification, and post-processing—as comprehensively summarized by Ünsalan and Boyer [5].

In terms of crop identification, Foerster *et al.* [3] propose to extract *spatio-temporal profiles* comprising *normalized difference vegetation index (NDVI)* information from LANDSAT-ETM satellite data for *maximum-likelihood (ML)* classification. In their experiments, the authors were able to classify twelve individual crop classes distributed over a 14 000 km<sup>2</sup> large study area in north-east Germany. In a comparable manner, Matton *et al.* [6] identify crops by statistical features derived from NDVI values and classify them by *K-means* and ML classifiers. They utilized LANDSAT-7 and SPOT-4 observations acquired from eight test regions distributed over the entire world. Similarly, Valero *et al.* [7] use *randomized decision forests (RDF)* on statistical features derived from several spectral indices from SENTINEL 2A data.

While these aforementioned approaches do not retain the sequential consistency of multi-temporal observations, *hidden Markov models (HMM)* or *conditional random fields (CRF)*—as, for instance, proposed by Siachalou *et al.* [8], and Hoberg *et al.* [9], respectively—can, to some extent, model the temporal order of sequential data inputs. Both approaches use a combination of *very high resolution (VHR)*

and *moderate resolution* satellite images on a short temporal series of observations. While Siachalou *et al.* [8] concentrated on eight crop classes in a relatively small area of interest, Hoberg *et al.* [9] classified four more broadly defined land cover classes in their experiments, with crops being condensed to the class *cropland*.

Kernel-based methods have also been evaluated for multi-temporal classification, with Camps-Valls *et al.* [10] introducing a family of kernels to utilize temporal contextual and multisensor information. The proposed kernels were tested both on real optical LANDSAT 7 and synthetic data. The cross-information kernel was found to be best in general, but a simple summation kernel performed similar, as pointed out by Mountrakis *et al.* [11].

Along with the great success of deep learning methods, *convolutional neural networks (CNN)* came into the focus of the LCC research community. Nevertheless, to date most approaches do not follow the end-to-end training paradigm, which is inherent to deep learning, but rather resort to networks pre-trained to different computer vision problems and fine-tune them to specific LCC application scenarios. Most commonly, authors propose to rely on *CaffeNet* [12] (as an extension of *AlexNet* [13]), *GoogLeNet* [14], or *ResNet* [15] architectures to extract features to be categorized into crop classes by *support vector machine (SVM)* or *softmax* classifiers [16–18]. Castelluccio *et al.* [16], for instance, reported experimental results showing that training CNNs entirely from scratch using remote sensing data—*i.e.*, the *UC Merced* [19] database—resulted in worse performance compared to fine-tuning or reusing of pre-trained features. This is most likely due to a limited amount of annotated data available for optimizing the millions of parameters involved.

Methodically most similar to our approach, Lyu *et al.* [20] recently proposed to use *recurrent neural networks (RNNs)* [21] and *long short-term memory (LSTM)* networks [22] to analyze remote sensing imagery but, in contrast to our scenario, for the sake of binary and multi-class change detection.

## 3. Approach

As previously set out, we aim to model the sequential change of crop phenology during the growth season to assist further land cover classification. Inspired by recent advances in machine learning and computer vision, we propose to employ *long short-term memory (LSTM)* networks [22] to learn vegetation grammar patterns based on sequential observations. In our experiments, we rely on SENTINEL 2A satellite data acquired over the entire growth period in form of *bottom-of-atmosphere* reflection information.

### 3.1. Neural Network Architectures

The impressive success of recent *deep learning* systems was predominantly achieved by feed-forward neural network

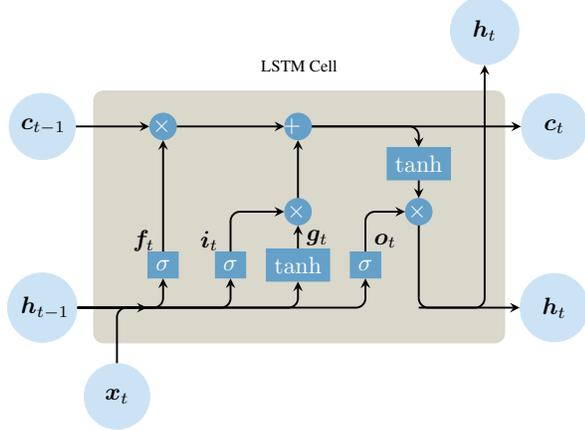


Figure 2. By augmenting standard *recurrent neural networks* (RNNs) by forget gates  $f_t$ , input gates  $i_t$ , modulation gates  $g_t$ , and output gates  $o_t$ , *long short-term memory* (LSTM) networks are capable of controlling the amount of stored information from previous observations  $x_0, \dots, x_{t-1}$ . (Figure adapted from [colah.github.io/posts/2015-08-Understanding-LSTMs](https://colah.github.io/posts/2015-08-Understanding-LSTMs))

architectures with strict sequential data propagation

$$\mathbf{h} = \sigma(\mathbf{W}_{\text{data}} \cdot \mathbf{x} + \mathbf{b}) \quad (1)$$

from input data vectors  $\mathbf{x} \in \mathbb{R}^n$  to hidden state vectors  $\mathbf{h} \in \mathbb{R}^m$  as a combination of affine transformations  $\mathbf{W} \in \mathbb{R}^{n \times m}$ , biases  $\mathbf{b} \in \mathbb{R}^m$ , and non-linear activation functions  $\sigma : \mathbb{R}^m \mapsto \mathbb{R}^m$ , *e.g.* sigmoid and hyperbolic tangent functions or *rectified linear units* (ReLU). While this design is favorable for processing individual uncorrelated data items  $\mathbf{x} \in \mathbb{R}^n$ , the use of time-dependent data  $\mathbf{x}_t \in \mathbb{R}^n, 0 \leq t < T$ , requires the network to perform context-sensitive data processing.

A specialization of these fully-connected or convolutional neural networks are *recurrent neural networks* (RNNs) [21]. RNNs are capable of storing information and intermediate results in order to influence the processing of future observations while being able to process data with unlimited sequence lengths. This functionality is realized by augmenting the layer-wise non-linear mapping

$$\mathbf{h}_t = \sigma(\mathbf{W}_{\text{data}} \cdot \mathbf{x}_t + \mathbf{W}_{\text{state}} \cdot \mathbf{h}_{t-1} + \mathbf{b}) \quad (2)$$

by the state vector  $\mathbf{h}_{t-1}$  derived from previous data  $\mathbf{x}_{t-1}$ . Similar to the research field of *natural language processing* (NLP), where RNNs yielded broad success, remote sensing handles data with evident spatio-temporal dependencies.

Inducing a further level of complexity, LSTM networks [22], as visualized in Figure 2, are able to regulate the amount of intermediate data to be stored by adding several controls, *i.e.*

$$\begin{aligned} f_t &= \sigma_f(\mathbf{W}_{\text{data}}^f \mathbf{x}_t + \mathbf{W}_{\text{state}}^f \mathbf{h}_{t-1} + \mathbf{b}^f) && (\text{forget gate}), \\ i_t &= \sigma_i(\mathbf{W}_{\text{data}}^i \mathbf{x}_t + \mathbf{W}_{\text{state}}^i \mathbf{h}_{t-1} + \mathbf{b}^i) && (\text{input gate}), \\ g_t &= \sigma_g(\mathbf{W}_{\text{data}}^g \mathbf{x}_t + \mathbf{W}_{\text{state}}^g \mathbf{h}_{t-1} + \mathbf{b}^g) && (\text{modulation gate}), \\ o_t &= \sigma_o(\mathbf{W}_{\text{data}}^o \mathbf{x}_t + \mathbf{W}_{\text{state}}^o \mathbf{h}_{t-1} + \mathbf{b}^o) && (\text{output gate}). \end{aligned} \quad (3)$$

These gates influence the ability of LSTM cell to discard old information, to gain new information, and to use that information to create an output vector, respectively. The *cell state* vector

$$\mathbf{c}_t = \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \mathbf{g}_t \quad (4)$$

storing the internal memory is then updated using the *Hadamard operator*  $\odot$ , which performs element-wise multiplication, while the layer-wise hidden state vector

$$\mathbf{h}_t = \mathbf{o}_t \odot \sigma_h(\mathbf{c}_t) \quad (5)$$

is further derived from the LSTM output gate vector  $\mathbf{o}_t$ .

With these considerations in mind, the use of (convolutional) neural networks in general and RNNs and LSTM units in particular comes with manifold advantages for the motivating LCC scenario:

- i) Neural networks are capable of learning complex relations solely based on presented input data and corresponding labels (*i.e.*, end-to-end learning), superseding manual process modelling.
- ii) Due to information propagation along the observation sequence, RNNs and LSTMs are capable of learning temporal relationships between the data items.
- iii) At each point in time, classification decisions are based on relevant information from the entirety of all previous observations, extracted and stored in a globally optimal manner.
- iv) The classification process itself is robust to high-frequency temporal coverage—*e.g.*, by clouds, snow, *etc.*—, as long as these perturbations are adequately represented in the corpus of training data.

### 3.2. LSTM Networks for Temporal Vegetation Modelling

To address the original land cover classification problem, we propose a training and classification pipeline, as shown in Figure 3, to predict class probabilities based on a temporal sequence of observations. Each observation is expressed as an input vector  $\mathbf{x}$  consisting of

- i) the *day of observation*, encoded as a time stamp  $t_t \in \mathbb{R}^+$  normalized to the length of the year and

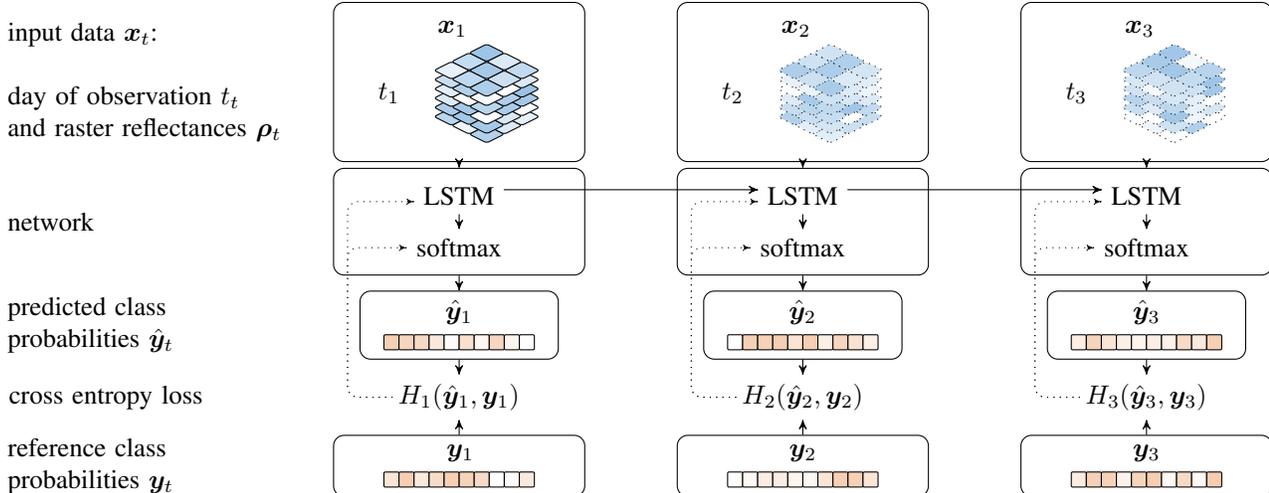


Figure 3. Information flow diagram of our proposed classification network based on LSTM modules during training and testing. Classification is performed at each timestep aided by information of previous observations, thus empowering LSTM networks to utilize temporal features.

ii)  $n_s$ -spectral reflection data  $\rho_t \in \mathbb{R}^{(k \times k) \cdot n_s}$  in an  $k \times k$  px neighbourhood around the *point of interest* (POI) to be classified.

A cascade of  $l$  LSTM layers with  $r$  cells per layer processes information based on the input of the current and of all previous observations. A final softmax layer transforms the LSTM output activation to class probabilities. At each training step, the cross entropy loss is calculated with respect to predicted and ground truth class probabilities. The calculated loss is backpropagated through the network layers as gradients, which in turn are utilized by *Adam* optimizer [23] to adjust the model weights.

## 4. Evaluation

In this section, we show how our approach is applied to a body of crop data and subsequently describe the performed experiments in Section 4.2

### 4.1. Data Material

For our experiments, a  $102 \text{ km} \times 42 \text{ km}$  study area in the north of Munich, Germany, has been chosen as area of interest (AOI), due to its homogeneous agricultural, geographical, and climate conditions. To monitor the growth season of 2016, we compiled a raster dataset of 26 SENTINEL 2A images acquired between 31<sup>st</sup> December, 2015 and 29<sup>th</sup> August, 2016 from the ESA SCIENTIFIC DATA HUB. The data is atmospherically corrected using the standard SEN2COR software package. In order to ensure comparability to the LANDSAT series, we selected the 10 m *ground sampling distance* (GSD) resolution bands (*i.e.*, 2 blue, 3 green, 4 red, 8 near-infrared) along with the 20 m GSD bands (*i.e.*, 11 short-wave-infrared-1, 12 short-wave-infrared-2) sampled to 10 m GSD by nearest-neighbour interpolation.



Figure 4. Area of interest (AOI) located in the north of Munich, Germany.

As ground-truth information, class labels for 137 k instances of the `fields` subset present in this AOI have been provided by the *Bavarian Ministry of Agriculture* (“*Bayrisches Staatsministerium Ernährung, Landwirtschaft und Forsten*”) in form of field geometry and names of cultivated crops. In total, this resulted in 19 *field* classes with at least 400 occurrences within the AOI selected for further evaluation. Particularly, these encompass the classes *corn*, *meadow*, *asparagus*, *rape*, *hops*, *summer oats*, *winter spelt*, *fallow*, *winter wheat*, *winter barley*, *winter rye*, *beans*, *winter triticale*, *summer barley*, *peas*, *potatoes*, *soybeans*, and *sugar beets*. The rejection class label *other* has been assigned if no field geometry was available.

Neural networks are usually trained on the body of training data in multiple epochs. Hence, it is advantageous to initially cast the input and output data to an appropriate format to ensure fast data retrieval and to avoid bottlenecks in data IO at training. For this reason, we derived a third

point dataset from the `field` and `raster` meta-datasets. We extracted 406 k points of interest (POI) following a regular  $100\text{ m} \times 100\text{ m}$  grid sampling scheme. Each of these POIs incorporate information of network input  $\mathbf{x}$  and labels  $\mathbf{y}$  in a  $30\text{ m} \times 30\text{ m}$  neighbourhood in the matrix dimensions required for the network. Input data  $\mathbf{x}$  for the network comprises the day of observation combined with raster reflections in a fixed  $3 \times 3\text{ px}$  neighbourhood. Label information  $\mathbf{y}$  was derived at the location of each pixel from the `field` dataset. We avoided hard class assignments by weighting multiple classes for POIs located at field borders. To account for coverages at single observations, *covered* classes—comprising *cloud*, *water*, *snow*, and *cloud shadow*—have been assigned based on the *scene classification mask* delivered by SEN2COR.

## 4.2. Results

We evaluated the performance of our proposed approach in two experiment lines.

First we analyzed the effect of different data sampling regimes on classification results, as to be reported in Section 4.2.1. After determining the optimal partitioning regime for our dataset, we trained multiple temporal and non-temporal models to be described in Section 4.2.2 and compared the performance in context of temporal features.

For this purpose, the network architectures were implemented in TENSORFLOW. The SCIKIT-LEARN PYTHON library was used to realize the SVM baseline and to calculate the classification metrics in Table 1. Grid search was performed within 8 hours on a NVIDIA DGX-1 server equipped with 8 TESLA P100.

### 4.2.1 Dataset Partitioning

Two main assumptions for the `training` and `testing` datasets need to be satisfied in order to evaluate the models independently from the respective datasets.

- i) Both datasets are independent from each other.
- ii) The class distributions in both datasets are sufficiently similar.

It is common practice to assign samples randomly to the respective dataset, which ensures that class distributions are similar, but assumes implicitly that the samples from each dataset are independent.

In order to evaluate these sampling effects, the dataset corpus was partitioned once into `train` and `test` datasets following three different policies in a 5:1 ratio:

**sample-wise random** Each data sample was *randomly* assigned to either dataset.

**block-wise random** The AOI was divided into blocks of  $3\text{ km} \times 3\text{ km}$ . Each block was then randomly assigned

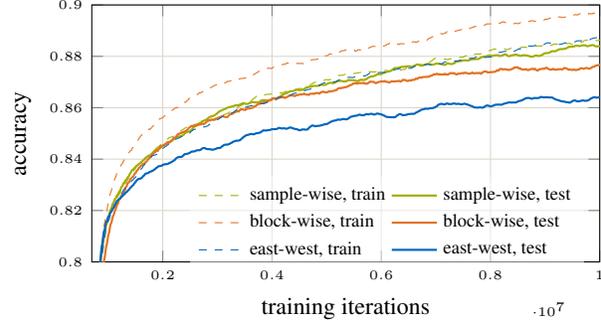


Figure 5. Crop classification performance of our LSTM network depending on the investigated training and testing partitioning schemes. While sample-wise random assignment (—) effected in overestimated accuracy, the block-wise random sampling (—) produced similar, yet more reliable results.

to the respective datasets, thus all contained POIs were then subsequently distributed.

**east-west** Data samples were assigned based on a north-south border line dividing the AOI into east and west subsets, which have been used as `test` and `train` datasets. Choosing such an east-west division ensures minimal length of the partitioning boundary, thus reducing spatial correlation.

Figure 5 shows the overall accuracies on `test` and `train` datasets of one LSTM network following these three separation schemes. For the case of sample-wise random separation, POIs in direct proximity to each other are likely to be assigned to the different datasets. However, these POIs might have been located at the same field plot and share characteristics, such as seeding or harvest dates. Thus, these POIs can be considered being dependent on each other, which subsequently resulted in overestimation of performance on `testing` data. Additionally, the difference between `train` and `test` data is minor, which might create the false impression of absent overfitting.

When the dataset was divided based on *east-west* separation, comparatively few points were located at close proximity. Hence, we can assume that both datasets are independent, as only few POIs at the division border can be located at the same field plot. The large spatial distance between `train` and `test` data, on the other hand, influenced the distribution of classes at each dataset. This relationship is apparent in Figure 6, where the class distributions in `training` and `testing` partitions are shown as logarithmic histograms. Class distributions after *east-west* separation can vary up to the point that no examples are available in the training dataset, such as the *hops* class in this experiment. This could likely have been caused by different regional environmental conditions, e.g., climate or soil quality, which encourage farmers to cultivate different crops based on local conditions.

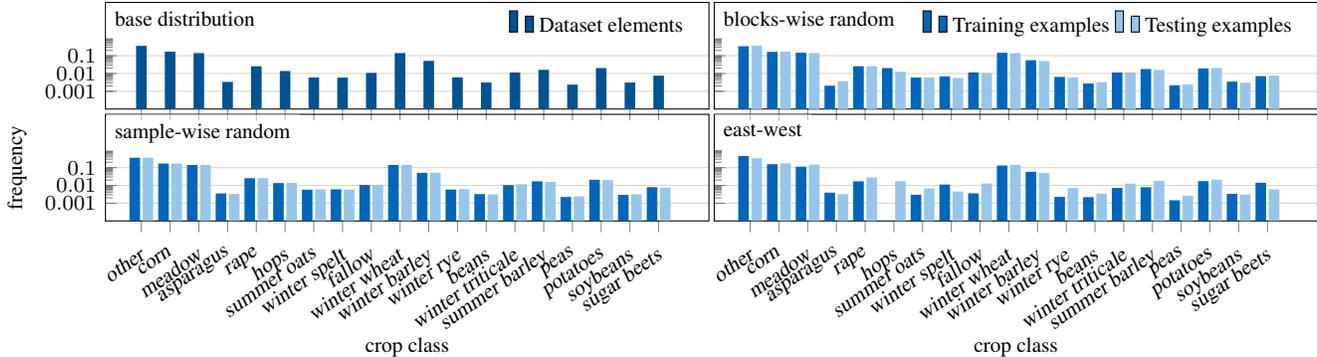


Figure 6. Histogram of class frequencies at each train and test data separation and the underlying base distribution. While sample-wise and block-wise random sampling did not change the class distributions significantly, east-west separation introduced major biases in class distributions

Additionally, the growth cycles of crops—observed as spectral signatures—could change gradually on a regional scale due to these environmental conditions. By introducing a large spatial distance between train and test datasets, the assumption of similar environmental conditions could be violated. This might have led to different growth behavior of the same crop type and, eventually, resulted in poorer classification performance, as one can observe in Figure 5 for the case of east-west partitioning. On the contrary, dividing the data body first in *blocks*, which are then assigned to the respective datasets, constitutes a compromise between both schemes. Because the number of POIs in direct spatial proximity is small due to the uniform block assignment, one can safely assume geographical independence. Additionally a margin between these blocks can be introduced to prevent POIs of the same fields from being assigned to different datasets. Simultaneously, the overall class distribution remained sufficiently comparable, as can be observed in the histogram corresponding to “blocks-wise random” in Figure 5.

#### 4.2.2 Influence of Temporal Features on Classifications

In order to evaluate the influence of temporal features on the classification, multiple temporal and non-temporal models have been compared in this experiment. To ensure independence of weights and hyper-parameters of the models, we partitioned the body of data into three datasets in a ratio 4:1:1. Based on the results of the previous experiment, we decided to partition the datasets using block-wise random sampling with a margin of 200 m between blocks. The training dataset was used to determine the network weights, while the validation dataset was used to select the optimal hyper-parameters. The training and validation datasets have been redistributed by 8-fold cross validation to maximize trained data and to average regional influences in class distribution. The final accuracy metrics were calculated

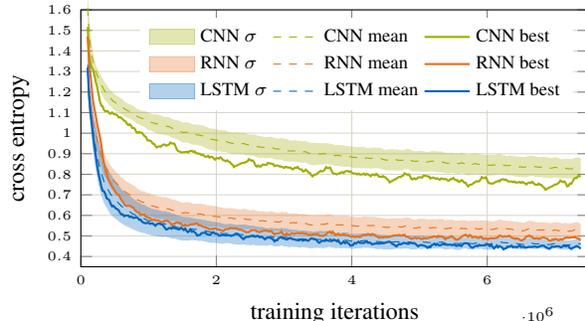


Figure 7. Training progress of 120 networks for each CNN, RNN, and LSTM architecture on the validation dataset. Realizing multi-temporal modeling, LSTM networks and RNN outperformed CNNs, while the influence of hyperparameters, indicated by minor standard deviation, had less influence on performance than the choice of architecture.

on the evaluation dataset, which remained untouched during all experiments.

**Neural Network Architectures** Three neural network architectures—*i.e.*, LSTM networks, RNNs, and CNNs—were evaluated by training 120 networks of each architecture with different hyper-parameter settings  $\theta_c = (l_c, r_c)$  for each architecture  $c \in \{\text{LSTM}, \text{RNN}, \text{CNN}\}$ . Hyper-parameters were chosen through a grid search, such that all combinations of the number of network layers  $l_c \in \{2, 3, 4\}$  and the number of cells per layer  $r_c \in \{110, 165, 220, 330, 440\}$  were tested. Even though each layer could be initialized with a different number of cells, which might benefit classification, we chose to keep the complexity of the grid search moderate and apply the same number of cells to each layer. To reduce overfitting on the presented training data, we added *dropout regularization* with keep probability  $p_{\text{keep}} = 0.5$ .

For each investigated network architecture, Figure 7 illus-

Table 1. Performance evaluation of our proposed LSTM-based method in comparison to standard RNNs and single-temporal baselines based on CNNs and SVMs. As *cover* classes (*i.e.*, *cloud*, *cloud shadow*, *water*, and *snow*) are usually comparatively easy to recognize, we restricted our evaluation to unbiased performance measures with respect to the remaining *field* classes. The accuracy metric was weighted by the frequency of samples in each class to avoid biases of the non-uniform class distributions.

Measure	Multi-temporal models						Single-temporal models					
	LSTM			RNN			CNN			SVM (baseline)		
	<i>all</i>	<i>cover</i>	<i>field</i>	<i>all</i>	<i>cover</i>	<i>field</i>	<i>all</i>	<i>cover</i>	<i>field</i>	<i>all</i>	<i>cover</i>	<i>field</i>
accuracy	90.6	93.6	<b>74.3</b>	89.8	92.9	72.9	89.2	93.7	64.3	40.9	87.4	31.1
AUC	98.1	97.5	<b>94.9</b>	97.8	97.0	94.1	95.1	97.0	84.7	87.1	97.6	81.6
kappa	77.6	55.6	<b>67.4</b>	76.1	53.0	65.6	66.2	56.3	44.0	38.2	83.4	27.3
precision	85.6	98.4	<b>78.4</b>	84.8	98.2	77.3	76.7	98.2	59.2	40.2	91.2	31.4
recall	84.4	92.5	<b>74.5</b>	83.4	91.8	73.0	76.8	92.7	57.2	40.9	87.4	31.1
f-score	84.6	95.3	<b>75.3</b>	83.6	94.9	74.0	76.1	95.3	56.7	40.3	88.9	31.1

trates the evolution of its mean and standard deviation of the validation loss during training.

Hyper-parameter settings  $\theta_{\text{LSTM}} = (4, 220)$ ,  $\theta_{\text{RNN}} = (4, 440)$ , and  $\theta_{\text{CNN}} = (3, 440)$  achieved best performance.

**Support Vector Machine Baseline** As a straight-forward mono-temporal baseline, we trained a SVM classifier with *radial basis function (RBF)* kernel  $\kappa_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|_2^2)$  on a balanced dataset of 3,000 data samples for each class. A grid search over the slack penalty  $C \in \{10^{-2}, \dots, 10^6\}$  and RBF scaling factor  $\gamma \in \{10^{-2}, \dots, 10^3\}$  was performed following a 10-fold cross validation scheme, on which the optimal hyper-parameters  $\theta_{\text{SVM}} = (C = 10, \gamma = 10)$  were determined.

**Comparison** Table 1 reports accuracy metrics of the best network of each architecture, along with the SVM baseline. All classifiers were capable of distinguishing *covered* classes well, with the SVM classifier especially achieving good accuracies. Moreover, *covered* classes and *field* classes were evaluated separately in order to obtain unbiased performance measures, as *field* classes are likely to develop characteristic behaviors over time. In contrast to *fields*, instances of *covered* classes, which appear based on high-frequent weather events, can be considered independent from long-term changes. Consequently, the multi-temporal LSTM networks and RNNs achieved better results on *field* classes compared to their mono-temporal competitors CNNs and SVM, as the former likely exploits the temporal changes of individual crops as classification features.

Figure 8 illustrates the kappa measure trends obtained for *field* classes of the best-performing LSTM, RNN, and CNN networks as functions over observations sequence lengths. While all three network architectures performed similarly in the first observations, after day of year 100, the classifica-

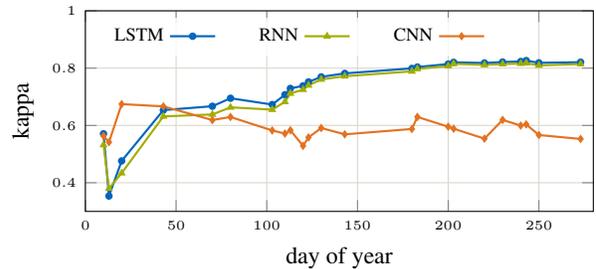


Figure 8. With increasing length of the observation series, multi-temporal LSTM and RNN networks learned to recognize the different crop types by means of their individual phenological vegetation cycles, while the mono-temporal CNN network did not benefit from these temporal characteristics.

tion accuracy of LSTM and RNN increased constantly with length of sequence. This performance increase of the temporal models around March and April coincides with the start of growth season in the AOI. After the winter period with mostly bare soil on all fields, crop-characteristic temporal changes are likely to occur with the vegetation period, which can potentially be utilized by LSTM and RNN networks, and thus increase the classification accuracy.

LSTM classification accuracies on the scale of individual crops can be observed from the confusion matrix shown in Figure 9. While some crop classes were classified with good confidence (*e.g.* *hops*, *rape*), some specific crops got confused more frequently. While a variety of reasons might have caused these confusions, some relationships can be explained by biological relations between crops. For instance, *triticale* is a hybrid of *rye* and *wheat*. These three crops are thus likely to share phenological events, as well as spectral appearances, which hinders distinctive classification. Other classes—such as *fallow*, *meadow*, and *other*—can not distinctively be defined and thus were misclassified with various other categories.

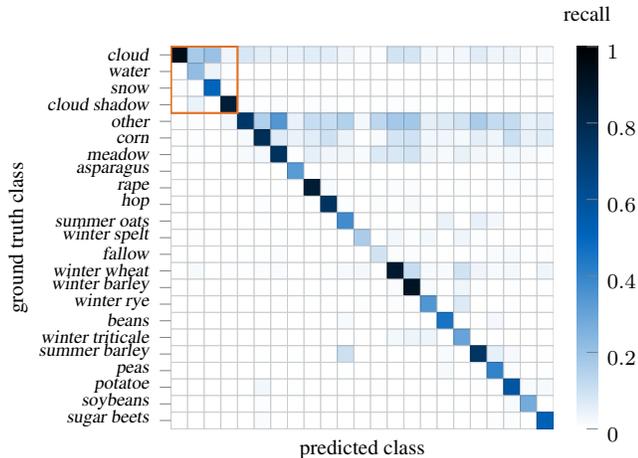


Figure 9. Confusion matrix reporting class-wise precision of the proposed LSTM-based approach. The orange-framed submatrix comprises cover classes, which have been excluded from further evaluation for the sake of unbiased performance assessment.

## 5. Discussion

Overall, as the performed experiments show, the temporal LSTM and RNN networks performed better than the non-temporal CNN and SVM models specifically on classes which inherit temporal characteristics, such as crops. Hence, we believe to have demonstrated that LSTM networks can utilize the temporal characteristics in the context of earth observation at the example of crop classes. Furthermore, our straightforward LSTM architecture showed superior classification accuracies on *field* data, in comparison to the state-of-the-art in per-plot crop identification [3, 8]. In terms of comparability with other research, it would have been favorable to train our networks on datasets of different approaches or to perform other classification techniques on our body of data. However, we were limited to reported performance on different datasets, due to restricted access of source code and data. To counteract this trend in future, we release our source code and the body data used for the training and testing of neural networks and the SVM baseline.<sup>1</sup>

As introduced in Section 2, the approach of Siachalou *et al.* [8] employing *hidden Markov models (HMM)* is comparable to our proposed method in terms of methodology. Their approach achieved good accuracies in experiments carried out on six crop classes using a combination of RAPID-EYE and LANDSAT imagery along with better kappa metrics compared to our LSTM network. However, the reported results could possibly have been skewed by the smaller AOI, which implies more homogeneous environmental conditions and farming practices. Moreover, their six crop classes possibly share a more orthogonal characteristic than our 19 *field* classes and thus could have been easier to distin-

<sup>1</sup>available at [www.lmf.bgu.tum.de/fieldRNN](http://www.lmf.bgu.tum.de/fieldRNN)

guish. Nevertheless, Markov-based approaches [8, 9] are strong competitors for utilization of temporal features for classification.

In terms of data characteristics, the dataset of Foerster *et al.* [3] is most comparable to ours, as their AOI is located in north-east Germany and contains a similar set of crops. Their more conservative classification approach is based on NDVI profiles extracted from LANDSAT-ETM imagery and additional agro-meteorological information. Our LSTM model achieved better performance in terms of both general accuracy and individual crop accuracy metrics, while still classifying a larger number crop classes and showing more flexibility in handling data by end-to-end learning.

After considering temporal features for the task of crop classification, results indicate that best accuracies are expected to be achieved by dynamic and self learning techniques, such as HMMs, CRFs, or deep learning, which is contrary to traditional hand crafted methods, such as spectro-temporal NDVI profiles.

## 6. Outlook and Further Work

Crop phenology changes based on environmental, and thus regional, conditions, such that a trained network in this vanilla form can not directly be applied to data of different regions. Nevertheless, an end-to-end learning scheme provides flexibility to introduce additional positional information, *e.g.*, temperature, length of day, or elevation. We have introduced, in a similar manner, the day of observation as input variable in our approach. Using additional regional information is believed to enhance the network capabilities to learn characteristics of crops at different regions. Moreover, training and evaluation at consecutive years would ensure separation and independence, as discussed in Section 4.2.1. In this work, we concentrated on the effects of temporal characteristics on the classification performance. Following this reasoning, we decided to fix the receptive field of our networks to  $3 \times 3$  px. Hence, we effectively restrict the end-to-end learning scheme in terms of spatial extent. Thus, the networks have not been able to adapt the extents of the involved perceptive field, as usually intended when using CNNs. In order to overcome this limitation, a CNN encoding pipeline could be attached in front of the LSTM cascade, thus mapping the input data in large-scale to the appropriate resolution in a non-linear manner. Thus, CNN encoders can potentially process the entire body of spectral data in different resolutions.

## Acknowledgements

We would like to thank NVIDIA for donating one TITAN X PASCAL graphics card and the *Bavarian Ministry of Agriculture* for providing information of crop cultivation at excellent geometric and semantic accuracy.

## References

- [1] H. Carrão, P. Gonçalves, and M. Caetano, "Contribution of multispectral and multitemporal information from modis images to land cover classification," *Remote Sensing of Environment*, vol. 112, no. 3, pp. 986–997, 2008. [1](#)
- [2] M. A. Friedl, D. K. McIver, J. C. Hodges, X. Zhang, D. Muchoney, A. H. Strahler, C. E. Woodcock, S. Gopal, A. Schneider, A. Cooper *et al.*, "Global land cover mapping from modis: algorithms and early results," *Remote Sensing of Environment*, vol. 83, no. 1, pp. 287–302, 2002. [1](#)
- [3] S. Foerster, K. Kaden, M. Foerster, and S. Itzerott, "Crop type mapping using spectral-temporal profiles and phenological information," *Computers and Electronics in Agriculture*, vol. 89, pp. 30–40, 2012. [1](#), [2](#), [8](#)
- [4] J. B. Odenweller and K. I. Johnson, "Crop identification using landsat temporal-spectral profiles," *Remote Sensing of Environment*, vol. 14, no. 1-3, pp. 39–54, 1984. [2](#)
- [5] C. Ünsalan and K. L. Boyer, "Review on Land Use Classification," in *Multispectral Satellite Image Understanding: From Land Classification to Building and Road Detection*. Springer, 2011, pp. 49–64. [2](#)
- [6] N. Matton, G. S. Canto, F. Waldner, S. Valero, D. Morin, J. Inglada, M. Arias, S. Bontemps, B. Koetz, and P. Defourny, "An Automated Method for Annual Cropland Mapping along the Season for Various Globally-Distributed Agrosystems Using High Spatial and Temporal Resolution Time Series," *Remote Sensing*, vol. 7, no. 10, pp. 13 208–13 232, 2015. [2](#)
- [7] S. Valero, D. Morin, J. Inglada, G. Sepulcre, M. Arias, O. Hagolle, G. Dedieu, S. Bontemps, P. Defourny, and B. Koetz, "Production of a Dynamic Cropland Mask by Processing Remote Sensing Image Series at High Temporal and Spatial Resolutions," *Remote Sensing*, vol. 8, no. 1, pp. 1–21, 2016. [2](#)
- [8] S. Siachalou, G. Mallinis, and M. Tsakiri-Strati, "A Hidden Markov Models Approach for Crop Classification: Linking Crop Phenology to Time Series of Multi-Sensor Remote Sensing Data," *Remote Sensing*, vol. 7, no. 4, pp. 3633–3650, mar 2015. [2](#), [8](#)
- [9] T. Hoberg, F. Rottensteiner, R. Q. Feitosa, and C. Heipke, "Conditional random fields for multitemporal and multiscale classification of optical satellite imagery," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 53, no. 2, pp. 659–673, 2015. [2](#), [8](#)
- [10] G. Camps-Valls, L. Gómez-Chova, J. Muñoz-Marí, J. L. Rojo-Álvarez, and M. Martínez-Ramón, "Kernel-based framework for multitemporal and multisource remote sensing data classification and change detection," *IEEE Transactions on Geoscience and Remote Sensing (TGRS)*, vol. 46, no. 6, pp. 1822–1835, 2008. [2](#)
- [11] G. Mountrakis, J. Im, and C. Ogole, "Support vector machines in remote sensing: A review," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 66, no. 3, pp. 247–259, 2011. [2](#)
- [12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional Architecture for Fast Feature Embedding," in *International Conference on Multimedia (ICM)*. ACM, 2014, pp. 675–678. [2](#)
- [13] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems (NIPS)*, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., 2012, vol. 25, pp. 1097–1105. [2](#)
- [14] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper With Convolutions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9. [2](#)
- [15] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. [2](#)
- [16] M. Castelluccio, G. Poggi, C. Sansone, and L. Verdoliva, "Land Use Classification in Remote Sensing Images by Convolutional Neural Networks," *arXiv preprint arXiv:1508.00092*, pp. 1–11, 2015. [2](#)
- [17] F. Hu, G.-S. Xia, J. Hu, and L. Zhang, "Transferring Deep Convolutional Neural Networks for the Scene Classification of High-Resolution Remote Sensing Imagery," *Remote Sensing*, vol. 7, no. 11, pp. 14 680–14 707, 2015.
- [18] G. J. Scott, M. R. England, W. A. Starms, R. A. Marcum, and C. H. Davis, "Training Deep Convolutional Neural Networks for Land-Cover Classification of High-Resolution Imagery," *IEEE Geoscience and Remote Sensing Letters (GRSL)*, vol. 14, no. 4, pp. 549–553, April 2017. [2](#)
- [19] Y. Yang and S. Newsam, "Bag-Of-Visual-Words and Spatial Extensions for Land-Use Classification," in *SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2010, pp. 270–279. [2](#)
- [20] H. Lyu, H. Lu, and L. Mou, "Learning a Transferable Change Rule from a Recurrent Neural Network for Land Cover Change Detection," *Remote Sensing*, vol. 8, no. 6, pp. 1–22, 2016. [2](#)
- [21] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990. [2](#), [3](#)
- [22] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997. [2](#), [3](#)
- [23] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014. [4](#)