Petteri Nevavuori

Title
Subtitle

The series, ISBN and ISSN numbers are added on this page in the library.

Dedicated to those writing dedications.

Seriously, **have** FUN with it.

PREFACE/ACKNOWLEDGEMENTS

Preface or acknowledgements here.

ABSTRACT

Abstract text here.

TIIVISTELMÄ

Other abstract text here, e.g. in Finnish.

CONTENTS

1 Introduction 17
1.1 Objectives 18
1.2 Publications and Author's Contribution 20

2 Data-based Smart Farming 23
2.1 Precision Agriculture and Smart Farming 24
2.1.1 Crop yield prediction 26
2.2 Data sources 27
2.2.1 Low-altitude unmanned aerial vehicles 28
2.2.2 High-altitude satellite systems 29
2.2.3 Weather data 30
2.2.4 Soil samplings 30
2.2.5 Topographical maps 31
2.2.6 Yield maps 31
2.2.7 Multisource inputs 31
2.3 Crop yield prediction 31

3 Spatiotemporal Deep Learning in Agriculture 33
3.1 Data arrangement 33
3.1.1 Preprocessing 33
3.1.2 Frame extraction 33
3.2 Spatiotemporal modelling 34
3.2.1 CNN 35
3.2.2 LSTM 35

3.2.3 CNN-LSTM 35
3.2.4 Convolutional LSTM 35
3.2.5 3D-CNN 35

4 Crop Yield Prediction with Deep Learning 37
4.1 Intra-field crop yield prediction 37
4.1.1 Single input to single target 37
4.1.2 Sequence of inputs to single target 37
4.2 Assessment of input data 37
4.2.1 Additional input sources 37
4.2.2 Satellite data reliability 37

5 Deep Learning Models as DSS Decision Engines 39
5.1 Decision Support System 39

6 Conclusions and Discussion 41
6.1 Discussion 41
6.2 Deep learning models as AI engines in a farming decision support system 42
6.3 Critique 42
6.4 Future outlook 43

References 45

Publication I 53

Publication II 65

Publication III 79

Publication IV 85

Publication V 105

List of Figures

List of Tables

List of Programs and Algorithms

ORIGINAL PUBLICATIONS

Publication I P. Nevavuori, N. Narra and T. Lipping. Crop yield prediction with deep convolutional neural networks. *Computers and Electronics in Agriculture* 163.June (2019). DOI: 10.1016/j.compag.2019.104859.

Publication II N. Narra, P. Nevavuori, P. Linna and T. Lipping. A Data Driven Approach to Decision Support in Farming. *Information Modelling and Knowledge Bases XXXI*. Vol. 321. 2020. DOI: 10.3233/FAIA200014.

Publication III P. Nevavuori, T. Lipping, N. Narra and P. Linna. Assessment of Cloud Cover in Sentinel-2 Data Using Random Forest Classifier. *International Geoscience and Remote Sensing Symposium (IGARSS)*. Accepted for publication. 2020.

Publication IV P. Nevavuori, N. Narra, P. Linna and T. Lipping. Crop Yield Prediction Using Multitemporal UAV Data and Spatio-Temporal Deep Learning Models. *Remote Sensing* 12.23 (2020). DOI: 10.3390/rs12234000.

Publication V P. Nevavuori, N. Narra, P. Linna and T. Lipping. Assessment of Crop Yield Prediction Capabilities of CNN usign Multisource Data. *Title missing*. Accepted for publication. 2021.

1 INTRODUCTION

This doctoral dissertation is about studying the applicability of novel machine learning methods with remote sensing data in the context of agricultural decision support systems (DSS) in precision agriculture [1] and smart farming [22]. Farmers have practiced precision agriculture for ages to optimize yield productions of their fields. Sources of intra-field variability were deduced by noting and exhanging annual observations and experimenting with interventions. However, both the observations and the conclusions drawn have been more or less based on intuition, rather than on objective data. From this emerges the need for data-driven decision making, i.e. smart farming, to aid the farmers in choosing the best actions to take to optimize crop cultivation [11]. The application of novel deep learning techniques, a subset of machine learning, has been on an increasing trend for the past few years in smart farming and precision agriculture application domains [13]. One of the key reasons for this progression is the abundant availability of sensor based data in terms of ground-based soil sensors, and low-altitude unmanned aerial vehicles (UAV) and high-altitude satellite systems [29]. Another factor is the open-access availability of other enviromental data, such as weather and land survey data. Thus, the use of remote sensing data to extract information with machine learning models for data-driven decision making has become more common in smart farming. Especially, the number of studies using deep learning techniques to perform agriculture related modelling tasks has steadily risen [10].

[Kuva S2 vs Drone]

Remote sensing data relevant to smart farming tends to be predominantly spatial in nature. This stems from the objects of interest - fields, forests and plots of land. Conventionally, open-access remote sensing data has been acquired from nationally operated multispectral satellite sources, such as Sentinel-S2 (ESA, Paris, France) or Landsat 8 (USGS, Reston, Virginia, USA). Satellite data, while spatial, is also temporal due to constant and frequent overflights over land and sea surfaces. Commercially

available UAVs have also been utilized [17]. While some UAVs come pre-fitted with quality RGB sensors, some systems are designed as platforms to which then desired sensor technology is to be mounted. In addition to the altitude from which the data is acquired, satellite and UAV data differ greatly in spatial resolution. While UAVs are often pre-fitted with modern RGB cameras and allow data capture resolutions well below 1 m^2/px, open-access satellite data is available at resolutions starting at 10 m^2/px (Sentinel-S2) and upwards. These data, satellite and UAV, are readily in image-like spatial format. Other field-related observational data, such as data from soil sensors or soil samplings, are often interpolated over plots of interest to generate image-like data in the form of spatial rasters.

The form of input data directly effects the selection of suitable data-based modelling techniques. Convolutional neural networks (CNN) [14, 15], a subset of neural network based deep learning techniques, excel with spatial data related tasks. These tasks include object recognition, imgae classification and image-based regression. Recently, multiple studies have been conducted with CNNs in the context of agriculture and smart farming [9]. The use of sequential models capable of extracting temporal features is also relevant with remote sensing data. Long short-term memory (LSTM) networks [4, 6], an implementation of sequential data utilizing recurrent neural networks (RNN) [19], have been shown to perform well in modelling tasks involving sequential data [8]. The LSTMs have to be coupled with CNNs to perform spatiotemporal modelling. Another way to tap into spatiotemporal data is to use three dimensional CNNs, where two dimensions are used for single point-in-time spatial inputs and the third dimension as the dimension of change between distinct spatial inputs [25].

1.1 Objectives

In the context of using field related remotely and manually gathered data, the specific objectives of this study are:

- to study the feasibility of machine learning models as analytical engines in a farming DSS and
- to assess the usability of data from multiple sources in agricultural deep learning.

The first objective is heavily centered around data based modelling with field related data. Excelling at complex decision making with fuzzy problems, humans are ill-equipped to derive causal and correlational relationships, whether linear or non-linear, from larger bodies of raw numerical data. Spatial data, such as RGB images of a field, consist of thousands of data points with multiple values associated to a single point. Spatial deep learning models, in the other hand, have been specifically developed to perform input-output mapping with spatial data. Due to the nature of these models, they require black-box optimization techniques to find the optimal combination of various hyperparameters. Hyperparameters are values, that have an effect on the training and the capabilities of the model. These values are, for example, the learning rate coefficient of the model's optimizing algorithm or the number of neurons, a calculation unit, within a layer of the layered deep learning architecture. Succsefully attaining the first objective requires also proper handling of input and target data samples. The data has to be both ingestable by the models, while the model's results have to be meaningful and interpretable by us humans. An additional key aspect is also the usability of the models post-development and post-validation. Having to do with usage and adoption, the usability of the models as a part of a bigger DSS has to be evaluated.

Generally, deep learning models benefit from feature-rich data. Being non-linear and layered models, they are optimized during training to find the most effective combinations of input and hidden features built from the input data to accomplish set performance goals. Data, however, incurs a resource cost on the modelling process. Firstly, the data acquisition has an effect on the overall feasibility of the modelling. UAV data, for example, requires manual operation in Finland due to legislation and regulations. Secondly, the contribution to model performance is not equal between distinct data sources. Yet another aspect of data is its quality, which itself might affect the general performance of the model and the system the model is used in. Thus, the data used in the modelling has to evaluated both in terms of feasibility and usability.

Together these two objectives help in addressing the larger question: Can deep learning and automated model pipelines be utilized to improve farming efficiency? For years, the number of farmers has been on a decline in Finland. With rather static number of field plots, the farms get bigger and are thus in need of better farm and process management tools. Manual, semi-automated and automated data acqui-

sition from various operational areas require data processing automation to provide actionable items in actionable time frame. Thus, this study is an attempt at studying whether data based modelling is beneficial for farm management and process optimization.

1.2 Publications and Author's Contribution

The publications for this dissertation fall into two categories. In the first category, the author did the majority of the work. In the other category, the work of the author was utilized in a study. The publications in the first category are [I], [III], [IV] and [V]. In these publications, the author alone was responsible for accumulating, preprocessing and preparing the data from various sources. The author carried out the work of developing, implementing and training the models presented in the publications. Model performance evaluation and comparison to the state-of-the-art research was also conducted by the author. In those publications the author did not partake, however, in manual data acquisition, such as operating the UAVs during the growing season. The author of this dissertation was also responsible for writing the majority of text in these publications. Publication [II] belongs to the second category. Only the model architecture, code and results of [I] were utilized as a case study in the report.

Intra-field crop yield prediction model [I] [IV]

Performing crop yield predictions from RGB image data requires using models capable of ingesting spatial data and deriving salient features from them. CNNs, being shift and space invariant artificial neural networks [32], are best suited for this modelling task. While open-access satellite data has already been utilized in crop related modelling, such as crop type classification and yield prediction, intra-field scale prediction with smaller fields common to Nordic countries requires images with higher resolution than what is currently available from the open-access satellite systems. More specifically, fields with side dimensions in just hundreds of meters are ill-represented by 10 m^2/px resolution satellite sources for intra-field modelling. Due to CNNs requiring input data having constant spatial dimensions, regular smaller frames are extracted from images of irregularly shaped fields. UAV-based orthomo-

saic images of crop fields have the data in a resolution high enough to allow for extracting image frames of fixed dimensions. Images taken at lower altitudes do not also suffer from clouds obstructing the view of the sensors.

As part of Mikä Data project carried out in the Data Analytics and Optimization research group of the Pori unit of Tampere University, several fields were imaged during the growing seasons of 2017-2019. The images of these fields were used to train models to perform frame-based crop yield prediction with single point-in-time [I] and time series [IV] image data. In this study, point-in-time is used as an expression to distinquish between temporally distinct inputs from temporal sequences of multiple inputs. Extensive tuning of optimizer related hyperparameters and architectural parameters is performed to find the best performing model composition in both modelling cases. The point-in-time model is based on a CNN, with its depth and onfiguration tuned to perform mapping of RGB image frames of crop fields to geolocationally matched yield data collected from yield mapping sensors during harvest. The time series model is evaluated from a selection of spatiotemporal deep learning model architectures: a CNN-LSTM, a convolutional LSTM and a 3D-CNN. The best performing model architecture for mapping time series of RGB image frames of crop fields to matching crop yield data was the 3D-CNN. While crop related modelling has been performed on larger scales at county-scale in USA [21] and China [7] and at nation-scale in Europe and Africa [20], field-scale UAV-based crop yield estimation for intra-field predictions is a novel contribution to the best knowledge of the author.

Remote sensing data evaluation [III] [V]

In addition to performing crop yield estimation with UAV remote sensing data acquired manually, the use of farming related sensor data, remotely and locally collected, for farming decision making is a topic of interest. As with any data, quality is one of the key interests. High altitude satellite-based earth observation suffers from occasional obstructions by cloud canopy. While Sentinel-S2 data products contain pre-calculated information about the possible presence of cloud cover, there's still work to do on the detection accuracy [2]. Using UAV RGB image data as ground truth for cloudless data of crop fields, a Random Forest ensemble decision tree was trained in [III] to perform pixel-wise cloudiness classification of Sentinel-S2 data.

Normalized difference vegetation index (NDVI) was calculated for UAV RGB and Sentinel-S2 true color RGB data and the difference used as an indicator for building the pixel-wise ground truth labels.

Another active area of research is combining data from multiple input sources to perform remote sensing data based modelling [5]. In [V], field-wise UAV RGB data was complemented with data from Sentinel-S2 satellites, manually collected soil samplings, soil's electrical conductivity, weather and topographical data. A CNN model configuration from [I] was then used as the baseline, as the performance had already been demonstrated with UAV RGB data. Training a baseline RGB-only model, several input data configurations were tested and evaluated to see which combination of input data sources would provide the best performance.

Decision support system for farming [II]

While applied machine and deep learning is an active area of research as of late [13], the research and development of user friendly decision support system platforms is crucial to deployment, and thus adoption, of developed models. In [V] a basis for such a platform was laid, with the focus being on the persistence and visualization of multisource spatial data of crop fields. Developed crop yield prediction models act as the artificial intelligence (AI) engine as a subsystem for the open-source Oskari-based (*www.oskari.org*, MIT & EUPL licensed) platform, generating refined predicted data for deriving actionable decisions during the growing season.

2 DATA-BASED SMART FARMING

The objectives of this thesis stem from the farmers' need to derive data-based farming decisions from data measured of their fields. While aggregated field-level data provides general guidelines, the actions and interventions are performed at the intra-field scale. The decisions also have to be made within an actionable time frame during the growing season. However, the data alone is not enough. As unmanned aerial system (UAS) overflights can be utilized provide frequent snapshots of distinct states of fields and crop growth, human-based information retrieval from spatially arranged numerical data is generally impossible. What is needed is an automated decision engine based on data-based machine learning techniques, capable of performing intra-field predictions using current state of crop development. Furthermore, this decision engine should be integrated in a holistic farming decision support system (DSS) to fully utilize the capabilities of modern sensors, connectivity and automatic data processing. This enables the farmers to make more informed decisions on what actions to take and in which parts of distinct fields.

In this chapter we will first review the relevant background and the current state-of-the-art smart farming and data sources in the context of crop yield prediction. While smart farming encompasses a broader farming context, from soil and water management to utilizing modern technology to optimize farming processes, we will constrain the discussion to the context of crop field management and crop yield estimation.

The chapter is constructed as follows. In the first section the current studies of data-driven smart farming are reviewed. This is to gain a proper view of the application context for machine learning models, which are discussed in Chapter 3. After that, data from distinct sources and the use thereof in agriculture-related modeling tasks is reviewed. Remote sensing is of particular interest, as it has been an active research area for a multitude of years already. Other data sources, such as soil and weather data, are also discussed. In addition to reviewing relevant studies, we will

also describe the data utilized in the studies related to this dissertation. In the last section of this chapter the modelling task of crop yield prediction is reviewed.

2.1 Precision Agriculture and Smart Farming

The technologization of the modern age farm has been a steady process, ongoing for several centuries. The first steps in this process were taken during the 18th century with important gradual developments in crop rotation and selective breeding techniques. After the World Wars, the farms were quickly mechanized and farming processes started to get more industrialized. Manual labor and animal work force with more effective machinery. As digital computation resources became more common via mainframe architectures starting at late 1960s, software products gained an immovable stance in agronomical counselling instutions and thus farming management practices. The introduction of the internet, developments in telecommunication, sensor and computer technologies enabled the farms gain increasingly detailed grasp of different areas of crop farming.

The current discussion in agricultural developments revolves around the concept of Smart Farming and digitalized agriculture [16, 18, 22, 29]. The key elements in Smart Farming revolve around data collection and utilization [12], data-based decision making [11], interconnectivity of cyber-physical systems [31], automation of farming processes [31] and improved management of farm processes [24]. Precision Agriculture is a term also present in current studies [24], but others have argued earlier that Smart Farming encompasses more than just addressing the field variability with modern technology and that Smart Farming is an extension of Precision Agriculture. In [22], Sundmaeker et al. describe Smart Farming as data-driven farming, where an interconnected system of machines, sensors and digital systems constitute the farm. Wolfert et al. share this view in [29], having smart devices controlling the farm as system. This is why we will continue the discussion using Smart Farming as the term for the concept of holistic digitalized cyber-physical farming.

Akin to how the post-war industrialization had a lasting effect on farming equipment and processes, the developments in information and sensor technologies have also steadily influenced and transformed the domain of agriculture. The introduction of digital computation first transformed the data handling and computation processes of agricultural experts and advisors, starting with punch hole cards and

progressing towards software applications [23]. In the recent years the developments in sensors, information technology (IT) systems and general adoption of digital farm management and decision support systems have further driven the transformation. This development is also reflected on recent studies. As discussed by Klerkx et al. in [12] in their review of digital agriculture, topics such as precision farming, internet of things (IoT), artificial intelligence (AI), robotics and overall transformative power of digital technologies have been a focus of an increasing number of agriculture related studies. Crop yield prediction has also seen a recent surge in the number of studies published year to year [13]. Furthermore, they view the digitalization of agriculture as shifting the focus of management towards data from various sources on-farm and coupling the farms to the larger food value chain. Data is viewed as a medium through which on-farmn and off-farm activities are optimized and new insights for further developing the processes are generated via modern data analysis and modelling techniques. These sentiments are shared also by Rose and Chilvers in [18], who see the digitalization of agriculture in the broader context of technologically re-scripting modern societies via the adoption of novel technologies and data-based management processes. In other words, they view digitalizing modern farms as a frontier of technological development, affecting societies and in part transforming them. According to them, multiple nation-scale entities have, thus, taken proactive measures for enforce the development and adoption of Smart Farming practices and technologies in the past few years, including the United Kingdom, Ireland, Greece, Japan and Australia.

One of the core elements of Smart Farming is data collection. Small and interconnected sensors, being more generally labelled as IoT-sensors, are utilized in tandem with sensors installed on farming equipment and machinery to produce a multi-source data stream from the farm. While novel insights tend to require using novel insight extraction with AI-related techniques, the accumulated data paints a holistic picture of the farm and its operations. This enables the farmers to base their decisions on measured data in timely and accurate manner [22]. Moreover, the developments in soil sensors planted in crop fields enable the farmers to remotely monitor their fields, which in turn allows them to make more informed decisions on actions to take [24]. Being a subject closely related to the IoT, performing data aggregation and analysis on-site via edge computing is another projected direction for agricultural cyber-physical systems [31].

Sensors, data and insights require effective management systems. A holisting agricultural management system addresses a farm's needs on multiple levels, such as accounting, traceability and on-farm process management. The management systems are also required to connect the farm to its stakeholders, such as consumers, public authorities and actors in the food value chain [24]. With the developments of the IT-sector in general, farm management solutions have also shifted from locally installed software to cloud-based services [31]. While this decouples the management solutions from physical devices on the farm to be used where needed, it also opens up new possibilities for data based decision making and insight generation with novel resource-intensive modelling techniques [18]. The adoption of Smart Farming practices makes the farm effectively a producer and manager of goods and operations related data. Being part of a larger agricultural ecosystem, the data generated on-farm is seen to benefit other instances, such as actors in the logistic chain and counselling institutions [11].

When Smart Farming is viewed as a holistic operating framework, equipment and independent systems add formidably to the complexity of the whole. There is a true need to further develop the integration of sensors, equipment, monitoring and management systems [22]. This calls for cooperation of business actors operating the in the domain of Smart Farming, IT operations being a focus of the development due to integrations. With working integrations, the benefits of accurate and timely automation can be reaped [31].

2.1.1 Crop yield prediction

As evident by now, Smart Farming encompasses a large variety of sub-tasks and smaller goals. The primary focus of this study is the task of crop yield prediction with data-based modelling techniques. Crop yield prediction is deemed one of the more challenging problems in the realm of Smart Farming. The output, the harvested crop yield, is affected by a variety of environmental, crop-related and farmer-induced factors. While novel data based modelling techniques, namely deep learning models, excel with multivariate and non-linear data, these distinct yield related factors are required to represented in the datasets used to train data based models [30]. In their review of machine learning based crop yield prediction, van Klompenburg et al. [13] observe that the data sources often present in crop yield prediction studies

include soil and crop information, climatological data, information about the nutrients and actions taken by the farmer. In addition to gathering data from multiple sources, it is also necessary to collect data across multiple years. As Filippi et al. discuss in [3], having the data cover larger time spans (*temporal coverage*) is deemed more important than having the field related data span larger areas (*spatial coverage*). A key aspect to using crop yield prediction in a Smart Farming DSS is to enable the farmer to decide on actionable items. Predicting the intra-field variability allows identifying underfperfoming areas in the fields [24]. With the increase of spatial resolution in predictions, the goals of Precision Agriculture are also easier to attain by focusing on distinct problem areas instead of treating the whole field in uniform manner.

2.2 Data sources

Remote sensing has played a significant role in advancing crop field monitoring during the past decades and is considered one the most important technolgies for Precision Agriculture and Smart Farming [27]. Especially, the publicly accessible high-altitude satellite systems, such as Sentinel (ESA, Paris, France) and Landsat (USGS, Reston, Virginia, USA) missions, have been a major catalyst in propelling remote sensing based agricultural research forward. While high altitude monitoring is good for observing larger areas, low-altitude unmanned aerial vehicles (UAV) unmanned aerial systems (UAS) are used to capture information in greater detail. While agricultural data is known to heterogeneous due to multiple factors at play [29], combining remote sensing data with data from additional sources generally improves modelling and analysis performance significantly [11]. Some data are human-generated, such as manual soil samplings or information about performed treatments. Other useful additional data sources include sensor-generated information, such as local weather stations, machine-equipped sensors and IoT-sensors scattered within crop fields' soil. Complementing remote sensing data with such automated sensors are seen to hold great potential for the developments towards automated Smart Farming ecosystems [28].

2.2.1 Low-altitude unmanned aerial vehicles

- UAVs used as data sources in agricultural data analysis [11] "Finally, some papers use air- planes and drones to achieve their goals, focusing on weeds' identifi- cation (Gutiérrez et al., 2008),"

[10]"Larger- scale observation is facilitated by remote sensing (Bastiaanssen et al., 2000), performed by means of satellites, airplanes and unmanned aerial vehicles (UAV) (i.e. drones), providing wide-view snapshots of the agricultural environments. It has several advantages when applied to agriculture, being a well-known, non-destructive method to collect in- formation about earth features while data may be obtained system- atically over large geographical areas."

[27] Overall this review is good and thorough on use of UAVs: "In the past, remote sensing was often based on satellite images [4,5] or images acquired by using manned aircraft in order to monitor vegetation status at specific growth stages. However, satellite imagery is often not the best option because of the low spatial resolution of images acquired and the restrictions of the temporal resolutions as satellites are not always available to capture the necessary images. ... The use of UAVs to monitor crops offers great possibilities to acquire field data in an easy, fast and cost-effective way compared to previous methods. UAV-based IoT technology is considered as the future of remote sensing in Precision Agriculture. UAVs' ability to fly at a low altitude results in ultra-high spatial resolution images of the crops (i.e., a few centimeters)."

[27]"popular application of UAVs in Precision Agriculture is Weed mapping"

[27]"UAVs are also used to monitor vegetation health. Crop health is a very important factor that needs to be monitored, as diseases in crops can cause significant economic loss due to the reduced yield and the reduction of quality."

[27]"Crop irrigation management is a very important area of application of UAV technologies in Precision Agriculture"

[27]"An application of UAVs in precision agriculture that is more rarely met is Crop spraying."

[26] "The future of precision agriculture lies upon modern technological advancements and remote sensing techniques using Unmanned Aerial Vehicles (UAV) and different kind of smart sensors."

[16] "Whilst remote sensing is a con-sistent research area, with proven applica-

tions in PA/PV when using imagery acquired by satellites and through manned airflights, it has more recently known significant and disruptive advances due to unmanned aerial systems (UAS) – that combine an unnamed aerial vehicle (UAV), a sensor as payload and a ground station, which has software to manage the flights (Pádua et al., 2017). Indeed, UAS enables non-in-vasive monitoring with spatial and temporal variability tailored to the crop and monitoring needs. Moreover, UAS have low-cost solutions, are very flexible platforms and enable the acquisition of different data through the use of various sensors."

[28] "With development IoT-based technolo- gies, unmanned aerial vehicles (UAV) have become an effec- tive tool for crop monitoring. Yang et al. [45] present a deep CNN for rice grain yield estimation. This method using remotely sensed images collected by UAV is able to make estimations at the ripening stage. Dyson et al. [46] integrated a radiometric index with terrain height images for segmenting crops and trees over the soil. High-resolution images col- lected by UAVs were used in the study."

[28] "Kerkech et al. [24] proposed deep lean- ing approaches for vine diseases detection using vegetation indices and colorimetric spaces, applied to images collected by UAV."

[28] "Given that high maneuverability, high mobility, and low maintenance cost, UAVs were used in studies related to almost all topics. In addition to being an effective tool, UAVs can contribute to the change from current practices ... UAV can be considered as an integral part of smart farming. ... Although UAVs are a key technological advance, they have some difficulties in their use in agriculture. Given their high power consumption during their flight, the flight time of UAV is quite limited [84]"

2.2.2 High-altitude satellite systems

[11] "Remote sensing has several advantages when applied to agriculture (Teke et al., 2013), being a well-known, non-destructive method to collect information systematically over very large geographical areas. A modern applica- tion of remote sensing in agriculture, as observed from the surveyed papers, is on the delivery of operational insurance products such as insurance from crop damage (de Leeuw et al., 2014; Global Envision, 2006), flood and fire risk assessment (de Leeuw et al.,

2014), or from drought and excess rain (Syngenta, 2010)"

[11]"On the other hand, data from high spatial resolution satellites like Landsat and SPOT have been used in support of local- and regional-scale applications requiring increased spatial detail (Ozdogan et al., 2010), such as farmers' decision making support (Sawant et al., 2016)."

[10]"From existing sensing methods, the most common one is satellite-based, using multi-spectral and hyperspectral imaging."

[28] "Multi- source satellite images are often used to capture specific plant growth stages. Several studies used deep learning for land productivity assessment and land cover classification. Kussul et al. [41] present a workflow for developing sus- tainable goals indicators assessment using high-resolution satellite data. Persello et al. [42] combined a full CNN with globalization and grouping to detect field boundaries. Zhou et al. [43] presented a deep learning-based classifier that learns time-series features of crops and classifies parcels of land."

2.2.3 Weather data

[11] "From the surveyed papers, relevant agricultural applications include weather forecasting (Schnase et al., 2014; Becker-Reshef et al., 2010)"

2.2.4 Soil samplings

Näytteet ja Veris MSP3

[11] "From the surveyed papers, relevant agricultural applications include ... soil and land (Barrett et al., 2014; Schuster et al., 2011)."

[24]"As an example, Pantazi et al. (2016) proposed the integration of high sampling resolution multi-layer data on soil and crop by using supervised neural networks to predict the spatial distribution of wheat yield with high accuracy."

[13] "The feature group "soil information" consists of the following variables: soil maps, soil type, pH value, cation exchange capacity, and area of production. Whether or not soil maps were used and the in- formation content of the maps differs among the different publications."

2.2.5 Topographical maps

2.2.6 Yield maps

2.2.7 Multisource inputs

[11] "Combining remote sensing with ancillary data (e.g. GIS data, historical data, field sensors, etc.) significantly improves the analysis performed, especially when it includes some form of prediction, i.e. crop identification (Waldhoff et al., 2012) or accuracy of distinguishing grasslands (Barrett et al., 2014)."

2.3 Crop yield prediction

3 SPATIOTEMPORAL DEEP LEARNING IN AGRICULTURE

3.1 Data arrangement

[10]"The large majority of related work (36 papers, 90%) involved someimage pre-processing steps, before the image or particular character- istics/features/statistics of the image were fed as an input to the DL model."

3.1.1 Preprocessing

[10]"Satellite or aerial images involved a combination of pre-processing steps such as orthorectification (Lu et al., 2017; Minh et al., 2017) calibration and terrain correction (Kussul et al., 2017; Minh et al., 2017) and atmospheric correction (Rußwurm and Körner, 2017)."

[10]"The most common pre-processing procedure was image resize (16 papers), in most cases to a smaller size, in order to adapt to the requirements of the DL model. Sizes of 256×256, 128×128, 96×96 and 60×60 pixels were common."

3.1.2 Frame extraction

[10]"Image segmentation was also a popular practice (12 papers), either to increase the size of the dataset (Ienco et al., 2017; Rebetez et al., 2016; Yalcin, 2017) or to facilitate the learning process by highlighting regions of interest (Sladojevic et al., 2016; Mohanty et al., 2016; Grinblat et al., 2016; Sa et al., 2016; Dyrmann et al., 2016a; Potena et al., 2016)"

3.2 Spatiotemporal modelling

[10]"imaging analysis is an important research area in the agri- cultural domain and intelligent data analysis techniques are being used for image identification/classification, anomaly detection etc., in var- ious agricultural applications (Teke et al., 2013; Saxena and Armstrong, 2014; Singh et al., 2016)."

[10]"The most popular techniques used for analyzing images include machine learning (ML) (K-means, support vector machines (SVM), artificial neural networks (ANN) amongst others), linear polarizations, wavelet-based filtering, vegetation indices (NDVI) and regression analysis (Saxena and Armstrong, 2014; Singh et al., 2016).Besides the aforementioned techniques, a new one which is recently gaining momentum is deep learning (DL) (LeCun et al., 2015; LeCun and Bengio, 1995)."

[10]"It is remarkable that all papers, except from Demmers et al. (2010,2012) and Chen et al. (2014), were published during or after 2015, indicating how recent and modern this technique is, in the domain of agriculture. More precisely, from the remaining 37 papers, 15 papers have been published in 2017, 15 in 2016 and 7 in 2015."

[24]"Over the past decade, machine learning techniques have been deployed across precision agriculture to provide more accurate solutions, mainly because of the capability to handle highly complex and non-linear agricultural problems (Liakos, Busato, Moshou, Pearson, & Bochtis, 2018; Morota, Ventura, Silva, Koyama, & Fernando, 2018). Machine"

[24]"Machine learning and data mining techniques are expected to be instrumental in meeting the challenges facing global agriculture, by taking advantage of big data. However, the collection and analysis of large, complex, heterogeneous data, coming from the variety of sources encountered in agriculture, cannot be accomplished with traditional machine learning methods such as linear regression."

[24]"However, in non-parametric approaches, the number of parameters is flexible, no assumption of the shape of the density is made, and the function's form is learned from the training data. The model learns from the data and can combine different data types. No prior choice of specific bands has to be made and all bands can be used to develop a model."

[13] "we show the distribution of applied deep learning al- gorithms in the identified papers list. The most applied deep learning algorithm is Convolutional Neural

Networks (CNN), and the other widely used algorithms are Long-Short Term Memory (LSTM) and Deep Neural Networks (DNN) algorithms."

[28] DL in agri has gained momentum recently: "The highest number of deep-learning-based agriculture-relevant papers on the database of the SCI appeared in 2019 (76) and there were no papers before 2016. The time trend analysis given in Table 1 ... The full list of those topics obtained from the analy-sis of 120 articles for the deep-learning-based agriculture domain is given in Table 2. Disease detection and plant classification are the most common topics, with 19 records, followed by land cover identification with 18 records, and precision livestock farming with 13 records."

3.2.1 CNN

3.2.2 LSTM

3.2.3 CNN-LSTM

[10]"Furthermore, some papers used features extracted from the images as input to their models, such as shape and statistical features (Hall et al., 2015),"

[13] has been used in agri-modelling tasks: "Sun et al. (2019) combined Convolutional Neural Net- works and Long-Short Term Memory Networks (CNN-LSTM) for soybean yield prediction. Khaki et al. (2020) combined Convolu- tional Neural Networks and Recurrent Neural Networks (CNN-RNN) for yield prediction. Wang et al. (2020) combined CNN and LSTM (CNN-LSTM) networks for the wheat yield prediction problem."

3.2.4 Convolutional LSTM

3.2.5 3D-CNN

[13] architecture has been used in crop yield prediction studies: "3D CNN: This network is a special type of CNN model in which the kernels move through height, length, and depth. As such, it produces 3D activation maps. This type of model was developed to improve the identification of moving, as in the case of security cameras

and medical scans. 3D convolutions are performed in the convolutional layers of CNN (Ji et al., 2012)."

4 CROP YIELD PREDICTION WITH DEEP LEARNING

4.1 Intra-field crop yield prediction

4.1.1 Single input to single target

4.1.2 Sequence of inputs to single target

4.2 Assessment of input data

4.2.1 Additional input sources

4.2.2 Satellite data reliability

5 DEEP LEARNING MODELS AS DSS DECISION ENGINES

5.1 Decision Support System

6 CONCLUSIONS AND DISCUSSION

Mallien kyky oppia merkittäviä piirteitä heterogeenisestä ja monimuuttujaisest datasta helpottaa käyttäjiä. Helpotus tulee siitä, ettei datan valinnassa tarvitse niin suuresti murehtia mahdollisesti mallintamiselle merkityksettömän datan raakaamisesta - mallit kykenevät oppimaan merkittävät yhdistelmäpiirteet syötedatasta ja jättämään merkityksettömän datan vähemmälle huomiolle. (Findrones2020)

Autokorrelaatio, miksi ongelma ja oliko ongelma

Muita ennustekohteita: puintiajankohta (voitaisiin hyödyntää myös muussa logistiikkaketjussa, mitenköhän kuivurihommat pelaa tähän? ainakin konealihankinta vois hyötyä)

6.1 Discussion

[12] "Digitalization in agriculture is thus expected to provide technical optimization of agricultural production systems, value chains and food systems. Furthermore, it has been argued that it may help address so- cietal concerns around farming, including provenance and traceability of food (Dawkins, 2017), animal welfare in livestock industries (Yeates, 2017) and the environmental impact of different farming practices (Balafoutis et al., 2017; Busse et al., 2015)."

[12]"The uptake inter- nationally of digital technologies in the past two decades has been most prevalent in agricultural sectors such as cropping and viticulture through precision farming technologies (Bramley, 2009), and to a lesser extent in animal-based farming (Borchers and Bewley, 2015; Eastwood et al., 2017a), and there are high expectations as regards its further diffusion and transformative potential (Rose and Chilvers, 2018; Shepherd"

6.2 Deep learning models as AI engines in a farming decision support system

[31] Where the computation should be located in a cloud-based computing system: "The subsystems of the edge layer make up the main operative control of the greenhouse and they are in charge of irrigation, climate, nutrition and auxiliary tasks, including alarms and energy management. At this layer, data fusion and aggregation is carried out to offload analytics functions usually performed in the cloud, given that the cloud part of the plat- form could serve a multitude of crops and users."

[12] "Managing of data acquired by a systematic collection of a vast array of heterogeneous sensors, continues to be as complex as the knowledge needed to make a decision on the practices of an agricultural process, as it requires processing in the various temporal and spatial dimensions of the acquired data. This task implies that data should become available in a suitable time window and format to be promptly viewed and stored, but also to feed decision-making support systems."

[12]"When a huge amount of data is collected over time, more advanced processing techniques should be employed. Big data, data analytics and artificial intelligence techniques are beginning to provide predictive insights and driving real-time decisions, innovating business processes. But, as stated in the review paper (Wolfert et al., 2017), big data and analytics applied to PA/PV are still at an early stage."

[18] "Furthermore, robotic technology could provide benefits to farming communities in compensation for lost labor, which is becoming a serious problem in the developing world as the population migrates to urban centers (Struik and Kuyper, 2017)."

[18]"concept of responsible innovation should underpin the fourth agricultural revolution; simply, ensuring that innovations designed to improve productivity and/or eco-efficiency also provide social benefits, meet human needs and are socially responsible."

6.3 Critique

[12] on the applicability of study's results, not every country is able utilize yet: "The article by Janc et al.(2019) on internet use among Polish farmers gives insights into

some of the basic elements needed to enable farmers to digitalize their practices. Janc et al. (2019) find that there is still a large 'digital divide' in terms of access and capabilities to use the internet."

[18] "Despite the potential benefits of a new technology revolution, the dominant techno-centric narratives associated with smart farming should be treated with caution (Whitfield et al., 2018). Technology is a double-edged sword because it has the potential to cause harm, as well as provide benefit (Stilgoe et al., 2013). ... The potential side- effects of smart technology like AI are being seriously considered now in policy (e.g., House ofLords., 2018)."

[18]"Rose et al. (2018a) argued that the requirement to use decision support tools would change, or "re-script," the ways in which farmers interacted with their land (see also Higgins et al., 2017 on how technology "orders" agricultural society)."

[18]"Wolfert et al. (2017) suggested that the emphasis on big data could further move decision- making power from the farmers into the hands of private companies who have control over such data (see also Carbonell, 2016)."

[18]"It is true that some smart technologies, such as precision agriculture, have so far been embraced with little societal "backlash," yet it is argued that large-scale use of AI, robotics, and other emergent innovations have the clear potential to cause unintended, unforeseen, and unwanted societal consequences. Indeed, Hartley et al. (2016) use the same precedent of the GM controversy to argue for the responsible governance of agricultural biotechnology."

[18]the development and adoption should be done in cooperation with farmers: "past experiences highlight the peril of ignoring risks, the danger ofbecoming "seduced" by innovation (Nordmann, 2014), and the fallacy of not seeking the views of publics in an effort to construct a shared vision of the future. ... Agricultural research is still dominated by top- down, non-inclusive approaches, and rarely includes relevant stakeholders, such as farmers, at an early stage (Macmillian, 2018)."

6.4 Future outlook

[12] I think that data collected on-site is mandatory to keep the digital twins and models accurate and relevant, as business isn't always as usual: "Can virtual models or digital twins, and Big Data re- place field experimentation?"

REFERENCES

[1] J. C. Bell, C. A. Butler and J. A. Thompson. Soil-Terrain Modeling for Site-Specific Agricultural Management. eng. *Site-Specific Management for Agricultural Systems*. Madison, WI, USA: American Society of Agronomy, Crop Science Society of America, Soil Science Society of America, 1995, 209–227. ISBN: 089118127X.

[2] R. Coluzzi, V. Imbrenda, L. Maria and S. Tiziana. A first assessment of the Sentinel-2 Level 1-C cloud mask product to support informed surface analyses. *Remote Sensing of Environment* 217 (Sept. 2018), 426–443. DOI: 10.1016/j.rse.2018.08.009.

[3] P. Filippi, E. J. Jones, N. S. Wimalathunge, P. D. Somarathna, L. E. Pozza, S. U. Ugbaje, T. G. Jephcott, S. E. Paterson, B. M. Whelan and T. F. Bishop. An approach to forecast grain crop yield using multi-layered, multi-farm data sets and machine learning. *Precision Agriculture* 20.5 (2019), 1015–1029. ISSN: 15731618. DOI: 10.1007/s11119-018-09628-4.

[4] F. Gers and J. Schmidhuber. Recurrent nets that time and count. *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks. IJCNN 2000. Neural Computing: New Challenges and Perspectives for the New Millennium*. IEEE, 2000, 189–194 vol.3. ISBN: 0-7695-0619-4. DOI: 10.1109/IJCNN.2000.861302. arXiv: arXiv:1011.1669v3. URL: http://ieeexplore.ieee.org/document/861302/.

[5] P. Ghamisi, B. Rasti, N. Yokoya, Q. Wang, B. Hofle, L. Bruzzone, F. Bovolo, M. Chi, K. Anders, R. Gloaguen, P. M. Atkinson and J. A. Benediktsson. Multisource and multitemporal data fusion in remote sensing: A comprehensive review of the state of the art. *IEEE Geoscience and Remote Sensing Magazine* 7.1 (2019), 6–39. ISSN: 21686831. DOI: 10.1109/MGRS.2018.2890023.

[6] S. Hochreiter and J. Schmidhuber. Long Short-Term Memory. *Neural Computation* 9.8 (1997), 1735–1780. ISSN: 0899-7667. DOI: 10.1162/neco.1997.9.8.1735. arXiv: 1206.2944. URL: http://www.mitpressjournals.org/doi/10.1162/neco.1997.9.8.1735.

[7] S. Ji, C. Zhang, A. Xu, Y. Shi and Y. Duan. 3D Convolutional Neural Networks for Crop Classification with Multi-Temporal Remote Sensing Images. *Remote Sensing* 10.2 (Jan. 2018), 75. ISSN: 2072-4292. DOI: 10.3390/rs10010075. URL: http://www.mdpi.com/2072-4292/10/1/75.

[8] R. Jozefowicz, W. Zaremba and I. Sutskever. An empirical exploration of Recurrent Network architectures. *32nd International Conference on Machine Learning, ICML 2015*. Vol. 3. 2015, 2332–2340. ISBN: 9781510810587.

[9] A. Kamilaris and F. X. Prenafeta-Boldú. A review of the use of convolutional neural networks in agriculture. *Journal of Agricultural Science* June (2018), 1–11. ISSN: 14695146. DOI: 10.1017/S0021859618000436.

[10] A. Kamilaris and F. X. Prenafeta-Boldú. Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture* 147 (2018), 70–90. ISSN: 0168-1699. DOI: https://doi.org/10.1016/j.compag.2018.02.016.

[11] A. Kamilaris, A. Kartakoullis and F. X. Prenafeta-Boldú. A review on the practice of big data analysis in agriculture. *Computers and Electronics in Agriculture* 143 (Dec. 2017), 23–37. ISSN: 0168-1699. DOI: 10.1016/J.COMPAG.2017.09.037.

[12] L. Klerkx, E. Jakku and P. Labarthe. A review of social science on digital agriculture, smart farming and agriculture 4.0: New contributions and a future research agenda. *NJAS - Wageningen Journal of Life Sciences* 90-91.October (2019), 100315. ISSN: 22121307. DOI: 10.1016/j.njas.2019.100315. URL: https://doi.org/10.1016/j.njas.2019.100315.

[13] T. van Klompenburg, A. Kassahun and C. Catal. Crop yield prediction using machine learning: A systematic literature review. *Computers and Electronics in Agriculture* 177 (Oct. 2020), 105709. ISSN: 01681699. DOI: 10.1016/j.compag.2020.105709. URL: https://linkinghub.elsevier.com/retrieve/pii/S0168169920302301.

[14] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard and L. D. Jackel. *Backpropagation Applied to Handwritten Zip Code Recognition*. eng. 1989.

[15] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner. Gradient Based Learning Applied to Document Recognition. *Proceedings of the IEEE* 86.11 (1998), 2278–2324. ISSN: 00189219. DOI: 10.1109/5.726791. arXiv: 1102.0183.

[16] R. Morais, N. Silva, J. Mendes, T. Adão, L. Pádua, J. A. López-Riquelme, N. Pavón-Pulido, J. J. Sousa and E. Peres. mySense: A comprehensive data management environment to improve precision agriculture practices. *Computers and Electronics in Agriculture* 162.March (2019), 882–894. ISSN: 01681699. DOI: 10.1016/j.compag.2019.05.028. URL: https://doi.org/10.1016/j.compag.2019.05.028.

[17] R. Näsi, N. Viljanen, J. Kaivosoja, T. Hakala, M. Pandžić, L. Markelin and E. Honkavaara. Assessment of Various Remote Sensing Technologies in Biomass and Nitrogen Content Estimation Using an Agricultural Test Field. *ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences* XLII-3/W3 (Oct. 2017), 137–141. ISSN: 2194-9034. DOI: 10.5194/isprs-archives-XLII-3-W3-137-2017.

[18] D. C. Rose and J. Chilvers. Agriculture 4.0: Broadening Responsible Innovation in an Era of Smart Farming. *Frontiers in Sustainable Food Systems* 2.December (2018), 1–7. ISSN: 2571581X. DOI: 10.3389/fsufs.2018.00087.

[19] D. E. Rumelhart, G. E. Hinton and R. J. Williams. Learning representations by back-propagating errors. eng. *Nature (London)* 323.6088 (1986), 533–536. ISSN: 0028-0836.

[20] R. Rustowicz, R. Cheong, L. Wang, S. Ermon, M. Burke and D. Lobell. Semantic Segmentation of Crop Type in Africa: A Novel Dataset and Analysis of Deep Learning Methods. *CVPR Workshops*. 2019, 75–82. URL: https://github..

[21] J. Sun, L. Di, Z. Sun, Y. Shen and Z. Lai. County-Level Soybean Yield Prediction Using Deep CNN-LSTM Model. *Sensors* 19.20 (Oct. 2019), 4363. ISSN: 1424-8220. DOI: 10.3390/s19204363. URL: https://www.mdpi.com/1424-8220/19/20/4363.

[22] H. Sundmaeker, C. N. Verdouw, J. Wolfert and L. Perez Freire. Internet of Food and Farm 2020. *Digitising the Industry*. 2016.

[23] J. Syväjärvi. *Reikäkorteista digiaikaan: Maatalouden Laskentakeskus Oy 30 vuotta, tietojenkäsittelyä 58 vuotta*. Suomen Maatalouden Laskentakeskus Oy, 2016, 105587–105609. ISBN: 978-952-93-7315-4.

[24] N. Tantalaki, S. Souravlas and M. Roumeliotis. Data-Driven Decision Making in Precision Agriculture: The Rise of Big Data in Agricultural Systems. *Journal of Agricultural and Food Information* 20.4 (2019), 344–380. ISSN: 15404722. DOI: 10.1080/10496505.2019.1638264. URL: https://doi.org/10.1080/10496505.2019.1638264.

[25] D. Tran, L. Bourdev, R. Fergus, L. Torresani and M. Paluri. Learning spatiotemporal features with 3D convolutional networks. *Proceedings of the IEEE International Conference on Computer Vision* 2015 Inter (2015), 4489–4497. ISSN: 15505499. arXiv: 1412.0767. URL: https://arxiv.org/pdf/1412.0767.pdf.

[26] A. Triantafyllou, P. Sarigiannidis and S. Bibi. Precision agriculture: A remote sensing monitoring system architecture. *Information (Switzerland)* 10.11 (2019). ISSN: 20782489. DOI: 10.3390/info10110348.

[27] D. C. Tsouros, S. Bibi and P. G. Sarigiannidis. A review on UAV-based applications for precision agriculture. *Information (Switzerland)* 10.11 (2019). ISSN: 20782489. DOI: 10.3390/info10110349.

[28] Z. Unal. Smart Farming Becomes even Smarter with Deep Learning - A Bibliographical Analysis. *IEEE Access* 8 (2020), 105587–105609. ISSN: 21693536. DOI: 10.1109/ACCESS.2020.3000175.

[29] S. Wolfert, L. Ge, C. Verdouw and M. J. Bogaardt. Big Data in Smart Farming – A review. *Agricultural Systems* 153 (2017), 69–80. ISSN: 0308521X. DOI: 10.1016/j.agsy.2017.01.023.

[30] X. Xu, P. Gao, X. Zhu, W. Guo, J. Ding, C. Li, M. Zhu and X. Wu. Design of an integrated climatic assessment indicator (ICAI) for wheat production: A case study in Jiangsu Province, China. *Ecological Indicators* 101.July 2018 (2019), 943–953. ISSN: 1470160X. DOI: 10.1016/j.ecolind.2019.01.059. URL: https://doi.org/10.1016/j.ecolind.2019.01.059.

[31] M. A. Zamora-Izquierdo, J. Santa, J. A. Martínez, V. Martínez and A. F. Skarmeta. Smart farming IoT platform based on edge and cloud computing. *Biosystems Engineering* 177 (2019), 4–17. ISSN: 15375110. DOI: 10.1016/j.biosystemseng.2018.10.014.

[32] W. Zhang, K. Itoh, J. Tanida and Y. Ichioka. Parallel distributed processing model with local space-invariant interconnections and its optical architecture. *Applied Optics* 29.32 (1990), 4790. ISSN: 0003-6935. DOI: 10.1364/ao.29.004790.

PUBLICATIONS

PUBLICATION

I

Crop yield prediction with deep convolutional neural networks

P. Nevavuori, N. Narra and T. Lipping

Computers and Electronics in Agriculture 163.June (2019)
DOI: 10.1016/j.compag.2019.104859

Computers and Electronics in Agriculture 163 (2019) 104859

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

Original papers

Crop yield prediction with deep convolutional neural networks

Petteri Nevavuori[b,*], Nathaniel Narra[a], Tarmo Lipping[a]

[a] Tampere University of Technology, Finland
[b] Mtech Digital Solutions Oy, Finland

ARTICLE INFO

Keywords:
Crop yield prediction
Convolutional neural network
Wheat
Barley
UAV
Multispectral
NDVI
Growth phase

ABSTRACT

Using remote sensing and UAVs in smart farming is gaining momentum worldwide. The main objectives are crop and weed detection, biomass evaluation and yield prediction. Evaluating machine learning methods for remote sensing based yield prediction requires availability of yield mapping devices, which are still not very common among farmers. In this study Convolutional Neural Networks (CNNs) – a deep learning methodology showing outstanding performance in image classification tasks – are applied to build a model for crop yield prediction based on NDVI and RGB data acquired from UAVs. The effect of various aspects of the CNN such as selection of the training algorithm, depth of the network, regularization strategy, and tuning of the hyperparameters on the prediction efficiency are tested. Using the Adadelta training algorithm, L^2 regularization with early stopping and a CNN with 6 convolutional layers, mean absolute error (MAE) in yield prediction of 484.3 kg/ha and mean absolute percentage error (MAPE) of 8.8% was achieved for data acquired during the early period of the growth season (i.e., in June of 2017, growth phase < 25%) with RGB data. When using data acquired later in July and August of 2017 (growth phase > 25%), MAE of 624.3 kg/ha (MAPE: 12.6%) was obtained. Significantly, the CNN architecture performed better with RGB data than the NDVI data.

1. Introduction

Development-minded farmers have practiced what is now known as precision agriculture long before the dawn of the computing age. They were able to deduce sources of field variability and the actions to take for trying to secure an enhanced level of crop yields. The farmers accomplished this by taking notes of their fields during growing seasons and harvest time operations and tried to figure out the best actions for the year to come based on the accumulated knowledge and experience. However, as studied by Wolfert et al. (2017), the increase in data-producing devices and sensors has been an on-going trend in agriculture having enabled the farmers to shift towards data-driven decision-making. This is commonly called smart farming. A comprehensive review of various objectives and techniques used in smart farming can be found in Kamilaris et al. (2017).

An important trend in smart farming is the use of remote sensing to facilitate the extraction of information relevant for data-driven decisions (Miyoshi et al., 2017; Matikainen et al., 2017). Remote sensing data can be acquired from satellites such as ESA's Sentinel-2A, for example. The problem with the satellite data is that if there is a cloud cover during the overflight of the satellite, no useful data are obtained. The spatial resolution of Sentinel imagery is at best 10 m, which is enough for many applications but too low to allow using texture-based information in the images. Satellite data contains predefined wavelength bands from both the visible and the Near Infrared (NIR) spectral regions. In satellite-borne sensors, designed keeping in mind agricultural applications, the spectral bands are optimized for the calculation of relevant indices such as the Normalized Difference Vegetation Index (NDVI), for example. The spatial and temporal resolution of satellite data will improve in years to come, however, cloud cover will remain an obstacle, especially in northern climate.

Using Unmanned Aerial Vehicles (UAVs), or drones, for data acquisition offers better spatial resolution, the data acquisition time can be selected by the user and the data can be acquired also in cloudy conditions. Spectral wavelengths can be selected by using appropriate camera; UAV-mountable RGB-NIR cameras are available at affordable price. The drawback is that the UAV has to be operated locally and managing the data and extracting relevant information requires highly specialized skills. As the variety of UAVs and UAV-mountable sensors is high compared to satellite-borne sensors, analysis frameworks and services based on UAV-borne data are not yet equally developed. In Näsi et al. (2017), extraction of information related to the biomass and nitrogen content of vegetation (barley and grass) in test fields using various modalities of remote sensing data (satellite/aircraft/drone using RGB/multispectral/hyperspectral sensors) has been considered.

Information relevant for decision making in agriculture can be

* Corresponding author.
E-mail addresses: petteri.nevavuori@mtech.fi (P. Nevavuori), nathaniel.narra@tuni.fi (N. Narra), tarmo.lipping@tuni.fi (T. Lipping).

https://doi.org/10.1016/j.compag.2019.104859
Received 17 April 2019; Accepted 13 June 2019

extracted from remote sensing data by means of machine learning. Traditional machine learning techniques involve feature extraction as an initial stage. Based on the features, different tasks such as crop classification, weed detection or yield prediction can be addressed. In Ruß (2009) several traditional machine learning techniques have been applied to the task of yield prediction. It is, however, often difficult to find optimal features and the ability of the traditional methods to learn from the data is limited. With advancements in computational technology, the development and training of novel multilayer algorithms has become feasible. These methods are commonly referred to as deep learning. Among the various deep learning paradigms, Convolutional Neural Networks (CNNs) have proved especially efficient in image classification and analysis. In case of CNNs no features need to be precalculated as the feature extraction operation is performed by the convolutional layers of the network and optimal features are obtained in the course of training. Due to this kind of structure, CNNs require large amounts of training data to converge. The advantage of CNNs compared to traditional machine learning methods in crop yield prediction is discussed, for example, in You et al. (2017). CNNs have been successfully applied to crop classification (Chunjing et al., 2017) and weed detection (Sa et al., 2017; Milioto et al., 2017).

In working towards an effective in-season crop yield predictor model for the northern climate, our effort in this preliminary study is to develop a CNN based deep learning framework using UAV-acquired multispectral data. RGB and NDVI images, representing patches of wheat and barley fields, are fed as input data to a CNN and training is performed to tune the network parameters. In addition to testing the usefulness of deep learning models for crop yield prediction in general, we also experiment with various setups and training schemes of the CNN model. Training a deep learning network is typically an iterative process as there is a substantial number of cross-related parameters to tune. We first select the most promising training algorithm from three candidates (see Section 3.1) and determine the optimal number of convolutional layers of the CNN. After that, we optimize the performance of the network in terms of regularization and parameters of the training algorithm. The optimized framework is evaluated using two types of input data (RGB and NDVI) and three patch sizes (10, 20 and 40 m).

2. Data and methods

2.1. Data acquisition

The nine crop fields selected for this study are located in the vicinity of the city of Pori (61°29′6.5″N, 21°47′50.7″E). The total area of the fields was approximately 90 ha. The main crops grown in the fields were wheat and malting barley, however the model was trained over the fields without making a distinction between the crop type.

Multispectral data were acquired from these fields during the growing season of 2017 (i.e., from June to August; see Table 1). The data were collected with a single Airinov Solo 3DR UAV equipped with Parrot's NIR-capable SEQUIOA-sensor. The images of individual spectral bands were stitched together to form complete orthogonal RGB and NDVI rasters of distinct fields using the Pix4D software.

The UAV data were organized into two sets according to the time of data acquisition to see if the phase of the growing season had an effect on predicting the yield from the input image. Growing phase here is defined as the percentage of total thermal time on the day of imaging. Thermal time for each day was calculated as the magnitude of daily average temperature above 5 °C. The temperature readings were downloaded from the Finnish Meteorological Institute. Beginning of July 2017 was chosen as the separating time point between the two data sets as the UAV data dispersed equally enough around that date. The data sets containing images only prior to July 2017 were labeled as *early* (growth phase < 25% of the total thermal time) and the remaining data as *late* (growth phase > 25% of the total thermal time). Details of the fields, crops, imaging dates and corresponding growth phases are listed in Table 1.

The field-wise image data were then processed using a sliding window to extract geolocationally matched pairs of input image frames (UAV data) and targets (yield data) of predefined size from all the fields. The step of the applied sliding window was chosen to be 10 m according to the resolution of Sentinel-2A satellite data considering the possibility of using satellite data as an additional input to the network in future studies. Image frames of sizes 10 × 10 m, 20 × 20 m and 40 × 40 m were considered. The resolution of the UAV data was 0.3125 m or 32 pixels per 10 m. The overall number of extracted frames according to crop fields is given in Table 2. The individual data frames were treated as independent inputs fed to the CNN models. The process of data preparation prior to and during training is illustrated in Fig. 1.

The harvest yield data was acquired during September 2017 using two distinct setups attached to the harvesters: Trimble CFX 750 and John Deere Greenstar 1. As the yield measurement devices produce an irregular set of data points with multiple attributes, the data had to be processed to be handled as rasters of field-wise yield from the viewpoint of the trainable network. The data points were first filtered according to (Tiusanen, 2017) to preserve only points corresponding to harvester speed between 2 and 7 km/h and yield between 1500 and 15,000 kg/ha. The filtering and generation of rasterizable vector files was done using the FarmWorks software. The field-wise vector data files were then rasterized by interpolating them using an exponential point-wise inverse distance algorithm. Yield values constitute targets the model is trying to predict during the training of the CNNs. Thus, yield values were also extracted using sliding windows similar to the UAV images to have geolocationally matching pairs of inputs and targets. Yield values were then averaged over the analysis window to obtain scalar target values. The histograms and statistics of yield values for point data as well as window-averaged data using three sample area window sizes (10 m, 20 m and 40 m) over all crop fields are given in Fig. 2. As can be expected, the larger the window, the more concentrated the yield values are around the mean.

For clarity, we also visualize several NDVI and RGB input images of the largest sample area window size (40 m) with their corresponding yields in Fig. 3 with the color bar corresponding to yield image value range. The images with similar identifiers are from the same location. However, the target for the network will be the mean of the yield values over the analysis window corresponding to the input area. It is also important to note that the network was trained separately for RGB and NDVI input images so that the possible misalignment between the two image sources does not affect prediction results. This kind of approach enables us to evaluate which one of the two input sources, RGB or NDVI, gives better prediction results.

2.2. Building the convolutional neural network

Convolutional neural networks, or CNNs, are deep learning models specialized in handling grid-like data. Such data can be images or rows of multi-column data. *Deep learning* refers to models composed of multiple layers. Generally, a model is viewed as deep if it has at least an input layer, one hidden layer and an output layer. The term *neural* on the other hand refers to the fact that originally the operation principle of artificial neural networks was taken from that of the brain, containing neurons as its basic building blocks. Compared to traditional feedforward neural networks, CNNs possess some special features making them extremely efficient in finding salient features within the data. Some of these features are:

1. exploitation of the convolution operation
2. post-convolution pooling
3. specific non-linear activation functions.

In the following we provide a brief description of these elements

Table 1
Details of crops and their varieties sown in each of the 9 fields in 2017. Thermal times for each crop variety are taken from a report published by Laine et al. (2017). Sowing dates and imaging dates are used to calculate the growth phase as a fraction of the total thermal time for the crop variety. Images with dates prior to 1st of July form the early data set and the remaining images the late one.

Field #	Size (ha)	Mean yield (kg/ha)	Crop (*Variety*)	Thermal time	Sowing date	Imaging date	Growth phase
1	5.96	5098	Wheat (*Zebra*)	1052	10 May	17 Aug	83%
2	10.26	6054	Barley (*Trekker*)	979.7	16 May	8 Jun	15%
						27 Jul	64%
3	2.97	8971	Barley (*Trekker*)	979.7	17 May	8 Jun	15%
						27 Jul	64%
4	13.05	4673	Barley (*RGT Planet*)	982.2	15 May	6 Jul	42%
5	4.66	6482	Barley (*Propino*)	981.4	15 May	15 Jun	22%
6	7.29	6884	Barley (*Propino*)	981.4	15 May	15 Jun	22%
7	10.92	7568	Barley (*Harbinger*)	976.3	24 May	6 Jul	36%
8	15.28	7585	Barley (*Trekker*)	979.7	18 May	1 Jun	10%
						13 Jul	49%
9	18.86	6991	Wheat (*KWS Solanus*)	1065	13 May	15 Jun	21%
						6 Jul	72%

Table 2
Number of data frames extracted from each field using frame sizes of 10 m, 20 m and 40 m. The number of frames decreases slightly with increasing frame size due to field edge effects.

Field #	10 × 10 m data frames	20 × 20 m data frames	40 × 40 m data frames	Mean data frame count
1	761	745	735	*747*
2	1102	1159	1150	*1137*
3	783	731	691	*735*
4	1494	1486	1454	*1478*
5	610	586	590	*595*
6	942	931	916	*930*
7	1240	1247	1224	*1237*
8	3736	3786	3812	*3778*
9	4556	4548	4520	*4541*
∑	**15224**	**15219**	**15092**	*15178*

with additional information on other key elements of CNNs such as batch normalization and regularization. We evaluated various setups of these CNN elements to find the best-performing algorithm and assess its performance in crop yield prediction.

2.2.1. Convolution operation

The convolution operation is the first of multiple transformations performed in a convolutional layer of CNNs. Generally, the convolution operation can be described as calculating the sum of products between a set of input values and values of a convolutional *kernel*, also called a *filter*. In CNN, the kernel values are trained to find optimal features from the point of view of the task to be solved (in our case, predicting crop yield). The operating principle of the kernel is depicted in Fig. 4 and the position of convolutional layers in the overall structure of the CNN used in this study can be seen from Fig. 7.

2.2.2. Batch normalization

While not a requirement for CNNs, the state-of-the-art is to apply batch normalization (Ioffe and Szegedy, 2015) as a constituent of deep learning model layers. Batch normalization is an optimization strategy

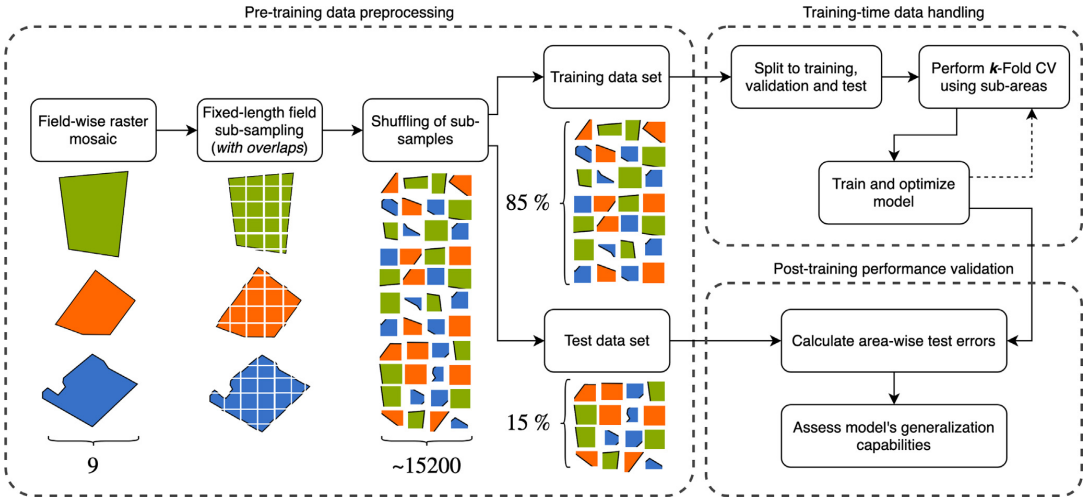

Fig. 1. All nine fields were first split to overlapping data frames of sizes 10 m, 20 m and 40 m. A dedicated holdout test data set was then built from 15% of shuffled data frames; these data were never presented to the model during training. The remaining 85% of data frames were then used for training the models with *k*-Fold Cross Validation. After the training phase of each model was completed, the test errors were calculated using the holdout test data set to validate the performance of the trained model.

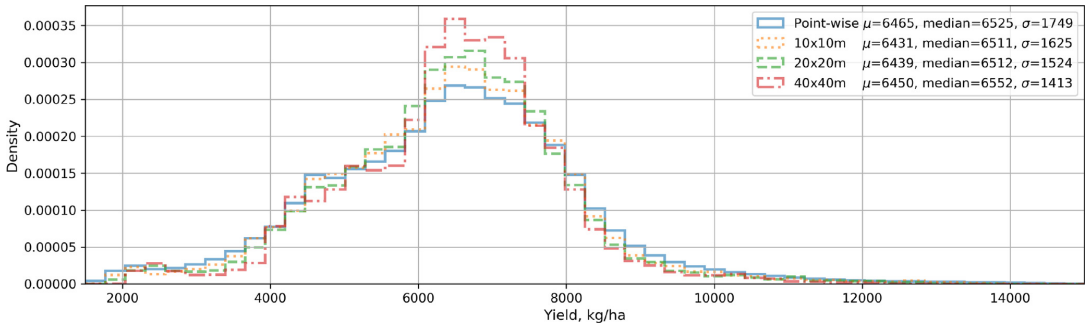

Fig. 2. Histograms and statistics of point-wise and window-averaged yield data. The histograms are normalized to probability densities to make point-wise graphs align with sliding window histograms count-wise. While sliding windows contain no-data points near field edges, only points containing data were taken into account.

for training deep models more efficiently. Batch refers to a subset of training data used for updating the model parameters (including kernel values) at a single iteration, albeit the term mini-batch is generally used to distinguish the whole data set (batch) from it's subset (mini-batch). It has been shown that normalizing the network layers for each batch (or mini-batch) of data stabilizes the learning, allowing to use higher learning rates and thus resulting in faster learning (Goodfellow et al., 2016). There are different implementations of batch normalization; the implementation used in the CNN of this study follows Eq. (1), where x is a mini-batch of activations, ϵ a non-significant constant to prevent numerical underflow, γ is the momentum and b is a layer-wise bias:

$$y = \frac{x - \mu_x}{\sigma_x + \epsilon} * \gamma + b. \tag{1}$$

2.2.3. Max pooling

The convolution operation is usually followed by pooling. Pooling means grouping of adjacent values using a selected aggregation function, which in our case was taking the maximum (hence max pooling) over the neighboring values within a predefined window. The step size of moving this window along the feature map is called *stride*. Pooling effectively diminishes the input image dimensions making the detected features more coarse and thus more robust to small variations (Goodfellow et al., 2016). The amount of dimension reduction is controlled by the stride parameter. The stride dictates how many applications of the pooling window are performed. An example of max pooling is given in Fig. 5 and the position of pooling in the overall structure of the CNN used in this study can be seen from Fig. 7.

2.2.4. Rectified linear units

A key element in any neural network is the layer-wise activation function of the neurons. A variety of activation functions have been designed, but the use of the rectified linear function in the activation units is the current standard for CNNs (He et al., 2015; Goodfellow et al., 2016). Activation units employing rectified linear functions are commonly referred to as ReLUs. The operating principle of this activation function is to allow only positive inputs to proceed linearly and is depicted in Fig. 6. We too use ReLUs as the activation functions in both the convolutional as well as the fully connected layers (see Fig. 7).

2.2.5. Fully connected layers

The convolutional layers of a CNN extract salient features from input images, i.e., factors with highest descriptive power regarding the data producing process. To utilize the learned features in a regression or a classification task, they have to be successfully mapped to a target value. This is performed typically by adding fully connected (FC) layers after the convolutional layers. The term *fully connected* refers to the principle that in these layers, each neuron (or unit) of the previous layer has a connection to each unit of the layer in question. Increasing the number of FC layers increases the capacity of the network to learn the mapping between the features and the target. It also increases the burden of optimization, as in FC layers the number of connections grows exponentially with the number of layers.

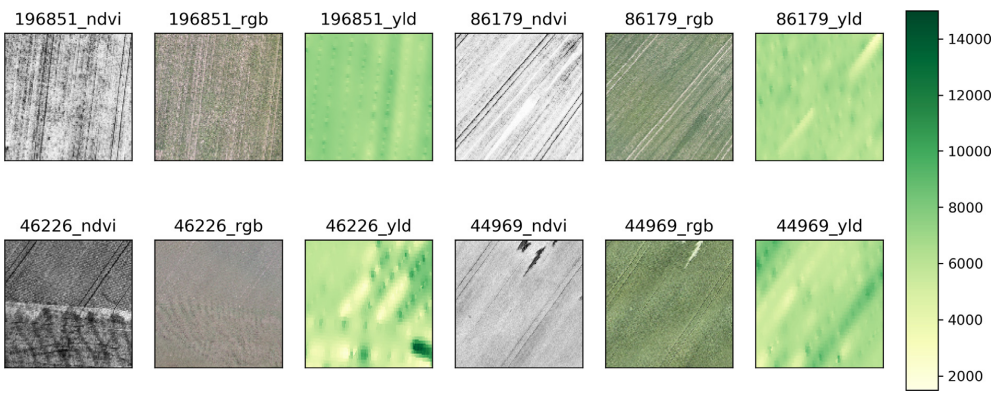

Fig. 3. Visualizations of NDVI and RGB input images and yield targets. The identification numbers above the images denote the distinct area from which the images were extracted.

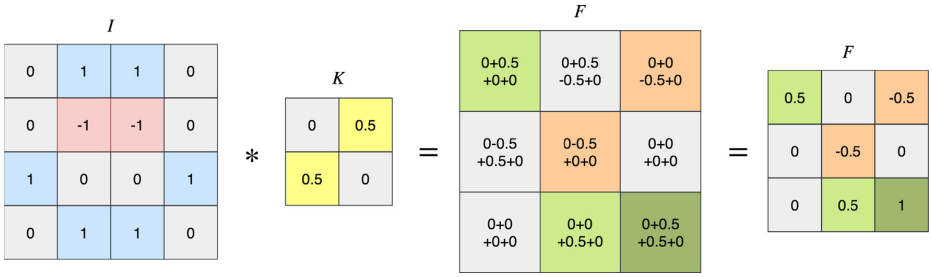

Fig. 4. The kernel K is applied to the input image I in a sliding window fashion. With each application, a sum of element-wise products is calculated and stored. After the kernel has been applied to the whole image, a complete feature map F is produced. A feature map indicates the result of detecting a kernel-specific feature in the input image.

2.2.6. Regularization strategies

Increasing the depth of a deep learning model allows it to learn more complex functions. This is also known as increased model's capacity (Goodfellow et al., 2016). When a model's capacity increases, it becomes more prone to overfitting to the training data in which case its ability to generalize (and, therefore, its performance on test data) deteriorates. This can be avoided with regularization, which effectively reduces the model's capacity diminishing the gap between training and test errors. Regularization is a comprehensive term for methods in machine learning that are used to lower the test error without focusing on training error.

In our model we make use of two distinct regularization strategies. First of the two is the L^2-penalty, also known as the weight decay. It diminishes the model's layer-wise parameters with each training iteration. When applied in conjunction with training by error back-propagation, the most relevant of the model's parameters retain their magnitude while non-relevant ones diminish. The second implemented regularization strategy is called early stopping. It is a robust meta-algorithm integrated into the training process to halt the training after n non-improving iterations. The hyperparameter n is called *patience* (Goodfellow et al., 2016).

2.2.7. Overall architecture

The basic architecture of the CNN implemented in this study follows closely the one reported by Krizhevsky et al. (2017). Their model performed extremely well in ImageNet Large Scale Visual Recognition Competition (Russakovsky et al., 2015) attaining top classification results in multiple categories. The general topology of our network is depicted in Fig. 7. The network was implemented using the PyTorch framework (Paszke et al., 2017). In our network we use non-overlapping pooling windows with pooling window size of 5 and a pooling stride matching the pooling window size. We also include the pooling function only in the first and the last convolutional layer. The reason for this is that at the lowest (i.e., in the case of 10 m ground resolution) our image size is 32 × 32 pixels and too many pooling operations would cause the data representation to collapse. This way our network is also scalable with respect to the number of layers. Regardless of the number of source image bands, our convolutional layers contain 64 kernels except for the last layer containing 128 kernels. Krizhevsky et al. (2017) incorporated two FC layers to the model with 2048 neurons per layer. We used similar number of layers with half the width, i.e., 1024 neurons per layer.

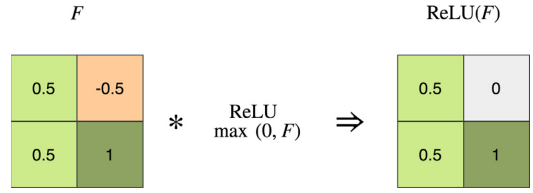

Fig. 6. An illustration of the effect of applying the rectified linear activation function to a pooled feature map.

2.3. Optimizing the network

Finding the optimal configuration of any deep learning network is an iterative process, where the model's parameters are initialized and tuned multiple times. The goal is to find a set of model's parameters (weights, biases, etc.) and hyperparameters (learning rate, optimizer coefficients, etc.) that in conjunction produce the best performance. The output of the iterative process is a single model usually performing best when compared to other models produced within the process. We used absolute error between the network output and the target value (i.e., crop yield values) as the performance measure. In machine learning, the best performing model is considered to be the one that generalizes well to previously unseen data. To measure the generalization performance across training instances, we extracted and reserved a subset of data as a holdout test set. This test data set was used outside of the training loop to ensure that the model never learned from it. With the rest of the data we performed k-fold cross validation using three folds per epoch. An epoch is a single complete iteration over the full training data set consisting of windowed image samples of all 9

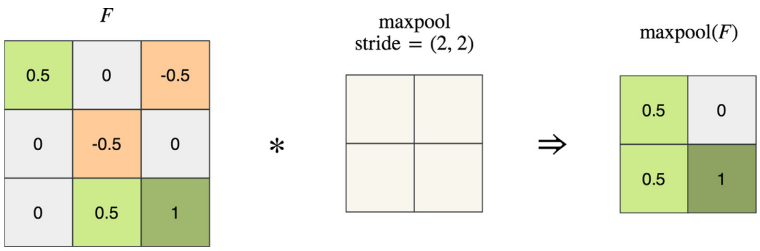

Fig. 5. An example of a simple application of max pooling, where the pooling is applied to a feature map F with pooling window size of 2 × 2 and a stride equaling the kernel size.

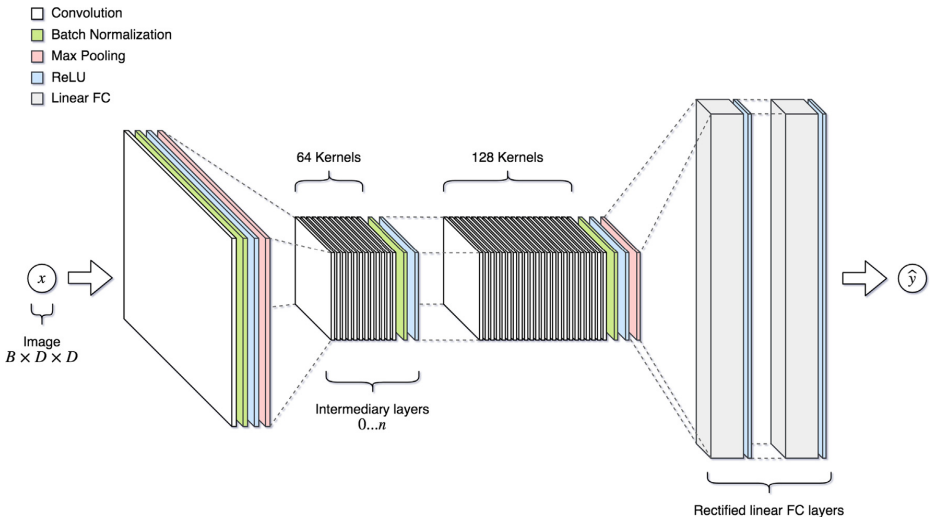

Fig. 7. The overall topology of the implemented CNN. Network's inputs can be single-band or multi-band images (B) with varying dimensions (D). The network has at least two convolutional layers accompanied with two fully connected layers. The depth of the network is controlled by the number of intermediary convolutional layers. The last convolutional layer has 128 kernels while the intermediary layers have 64 kernels. Max pooling is applied only in the first and last convolutional layers so that the size of the data representation stays consistent when network depth is varied.

fields.

The best training algorithm was evaluated among three options: Stochastic Gradient Descent with momentum (SGD-momentum) (Bottou, 1998), RMSprop (Hinton et al., 2014) and Adadelta (Zeiler, 2012). These training algorithms are suggested in Goodfellow et al. (2016) and they are also among the ones compared in Karpathy and Fei-Fei (2017). In a preliminary test, the three algorithms were tested for convergence by training the network for three epochs. Training was performed for each of the three data window sizes and each of the four sets of input data. The batch size was varied from 2^5 to 2^{10}. The worst performing algorithm was excluded and a second test performed on the remaining two by fixing the batch size to 128 (2^7) and training for 50 epochs, a number consistent across the training of almost every model.

The effect of the depth of the network on the performance was evaluated by training models with 4, 6, 8, 10 and 12 convolutional layers over 50 epochs per training session. The training was conducted for the NDVI and RGB images from early and late data sets and with all three input image dimensions using the previously selected training algorithm. At this stage, the best performing combination of - network depth, image type (NDVI or RGB) and window size - was selected based on error performance over the test data.

In the next step, the chosen training algorithm's hyperparameters (i.e., the learning rate and the past iterations' error correction adjustment) were tuned. In order to evaluate performance, benchmark models were created by initializing a model for each of the four data sets (i.e., early and late, RGB and NDVI). The hyperparameter values were searched over a coarse grid for values producing lowest test errors, followed by a more refined random search in the vicinity of the coarse minimum. Sensitivity of the network performance to initial values of the CNN parameters was also assessed.

In the last step, the hyperparameter combinations producing the best performance were used to test and tune the effect of regularization algorithms. Tuning of the weight decay coefficient (L^2 regularization) for early and late data sets was performed by searching over a coarse grid of values followed by refined search. Subsequently, the effect of early stopping was tested using values 10, 20, 30, 40 and 50 for the patience parameter (see Section 2.2.6).

3. Results

We measure the performance of the CNN by *mean absolute error*, i.e., the mean absolute difference between the true yield value and the CNN output (predicted value). This can also be called *loss*. We consider two different errors: the training error, obtained for the same data the network is trained with, and the test error, obtained for the data set aside for testing. The former one indicates how well the model is able to fit to the data, i.e., what is its capacity, while the latter one indicates how well the network is able to generalize to unseen data samples.

3.1. Selection of the training algorithm

Of the three training algorithms – Adadelta, SGD-momentum and RMSprop – the RMSprop showed poor convergence and was therefore ruled out from subsequent tests. Between the two remaining algorithms, Adadelta outperformed SGD-momentum and was chosen as the training algorithm for further experiments (see Table 3).

3.2. Depth of the network

In Fig. 8 the test and training errors for the three window sizes and for various networks depths are shown for the RGB data of earlier growth phase. The largest window (size 40 × 40 m) produced lowest test errors in majority of cases regardless of the network depth. The colored areas indicate gaps between training (lower bound of the area) and test (upper bound of the area) errors, also referred to as

Table 3
Lowest mean absolute test errors (kg/ha) observed among the three data window size configurations (10 m, 20 m and 40 m) with 50 epochs of training and a batch size of 128 samples for each source image type. Adadelta performed best with almost every source image configuration.

Optimizer	NDVI early	NDVI late	RGB early	RGB late
SGD with Momentum	1751.2	1183.7	1231.5	985.0
Adadelta	842.8	1165.1	836.2	989.5

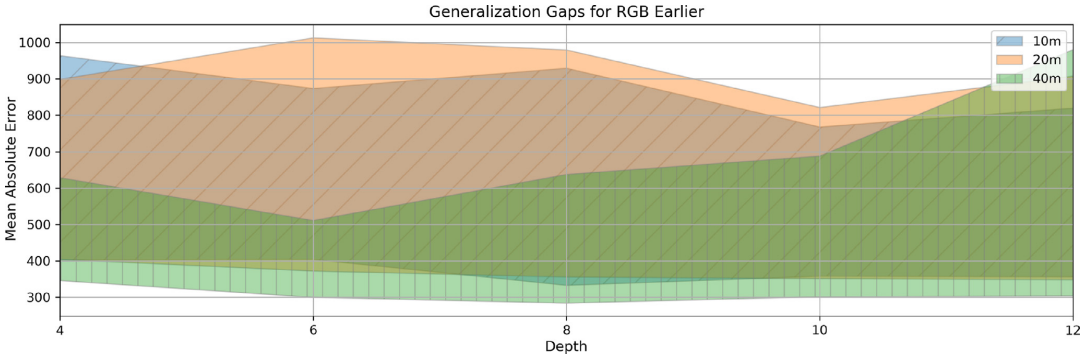

Fig. 8. The generalization gaps with early RGB images. The generalization gap is depicted as the difference between the training and the test errors. It shows how close the test error is to the effective capacity of the model, the training error. The lowest test errors (upper bound of the area) were achieved rather consistently with the source image window size of 40 × 40 m.

generalization gaps. The lowest test and training error combination is obtained with 6 convolutional layers. Also, the 40 m window and network depth of 6 convolutional layers result in the narrowest generalization gap.

3.3. Optimization of Adadelta hyperparameters

The hyperparameters of the chosen training algorithm, Adadelta, were tuned by considering the effects of the adaptive learning rate and the coefficient adjusting the effect of past iterations' error corrections (in the form of squared gradients) on learning. The latter is effectively similar to momentum, defining the magnitude by which the past affects the current learning process. The previous experiments were performed using default values for these hyperparameters, i.e., 1.0 for the learning rate, 0.9 for the coefficient for computing a running average of squared gradients, and 0 for the weight decay (see Table 4).

The initial grid search was conducted with hyperparameter values similar to those found in the original Adadelta research paper with an epoch limit of 50 compared to the original study's 6 epochs (Zeiler, 2012). For the early RGB data set, the optimal values were approximately 8×10^{-3} for the learning rate and 0.58 for the coefficient adjusting the effect of past iterations' error. For the late data set the respective values were 10^{-4} and 0.9. The effect of hyperparameter tuning on the performance of the network can be seen from the results in Table 4.

Table 4
The total improvement in test error compared to the benchmark model when using regularization and optimization of training algorithm hyperparameters. The benchmark models were trained with the early and late RGB image data with default parameters. Window size was 40 × 40 m. Errors are reported as mean absolute error (MAE) and mean absolute percentage error (MAPE). The best results are formatted in bold.

	RGB early		RGB late	
	MAE [kg/ha]	MAPE	MAE [kg/ha]	MAPE
Benchmark learning rate: 1.0 past err. coeff.: 0.9 weight decay: 0 patience: ∞	997.8	18.3%	1021.5	19.5%
with **Optimized Adadelta params.** learning rate (early/late): 0.008/0.0001 past err. coeff. (early/late): 0.58/0.9	546.2	9.6%	**624.3**	**11.4%**
and with **L²-regularization** weight decay: 0.001	558.4	9.4%	700.4	13.1%
and with **Early Stopping** patience: 50	**484.3**	**8.8%**	680.4	12.6%

3.4. Optimization of the regularization parameters

The CNN models using optimal hyperparameters for the Adadelta training algorithm were trained next with early and late RGB data sets of 40 × 40 m window to determine the effect of regularization on the prediction error and to tune the regularization parameters. The tuning of weight decay coefficient with grid search first and zoomed-in random search after that resulted in the optimal coefficient value of 10^{-3} for both data sets. The optimal patience values were around 50, again for both data sets. It was observed that the increase in patience increased the training time significantly. The selected patience value allowed the models for both data sets to converge in approximately 250 epochs. The effect of using the L^2-regularization alone and combined with early stopping can be seen from Table 4.

4. Discussion and conclusions

This study presents a training paradigm of a CNN based deep learning model for predicting wheat and barley yield. The results indicate that the best performing model can predict within-field yield with a mean absolute error of 484 kg/ha (MAPE: 8.8%) based only on RGB images in the early stages of growth (< 25% total thermal time). The model for RGB images at later growth stage returned higher error values (MAE: 680 kg/ha; MAPE: 12.6%). In searching for optimal performance, the input data window size (10 m, 20 m, 40 m), the data acquisition time (early vs. late) and data modality (RGB vs. NDVI) were varied. The 9 fields included in the study were imaged by a camera mounted to UAV and together taken as a source of > 10,000 input image frames covering a total area of 90 hectares. Network depth (i.e., the number of convolutional layers), the training algorithm and its hyperparameters as well as the CNN regularization scheme were also optimized. The lowest error was achieved using a network consisting 6 convolutional layers followed by two fully connected layers regularized with L^2-regularization coefficient of 10^{-3} and early stopping patience of 50. The optimizer was also tuned for the optimal value of the learning rate (8×10^{-3}) and the coefficient adjusting the effect of past iterations' error corrections (0.58). The results show that the lowest test errors were achieved with the largest data window size tested (40 m).

The training of any neural network is always influenced by the

combined randomness resulting from how the data is shuffled between cross validation folds, the optimization process and other factors. This in turn means that, while discrete error metrics produce a ranking across hyperparameter setups, slight variations between test errors can be attributed to the random nature of the optimization process as a whole. We optimized distinct models for early and late RGB data sets. The best performing model used RGB images from the early growing season and benefited from regularization. The model using the late RGB images didn't gain from added regularization, as the best performance was achieved during the tuning of the training algorithm (see Table 4).

In yield prediction, the shift from using traditional regression methods (Ruß, 2009) towards artificial neural network based methods (Chlingaryan et al., 2018) has resulted in improved performance (Jiang et al., 2004; Kaul et al., 2005). Among these ANN based studies, those using remote sensing image data to train their prediction models have achieved low prediction errors (≈ 5%). These models are specific to the crop types whose images they are trained with (e.g., soybean, wheat, rice). Jiang et al. (2004) working with satellite images reported an average relative winter wheat prediction error of 3.5%. The indices used for training the model were: NDVI, surface temperature, absorbed photosynthesis active radiation, water stress index and 10-year average crop yield. Bose et al. (2016) employed spiking neural networks to estimate winter wheat yield from satellite based NDVI images at the region level, achieving a best average relative error of 4.35%. In their recent work, You et al. (2017) leveraged advanced hybrid machine learning algorithms to achieve very low soybean yield prediction errors (3.19–5.65%) using only satellite images.

A commonality among these studies is the use of satellite imagery and large spatial scales of their analyses (region or county level predictions). Our study, in contrast, seeks to perform predictions at the intra-field scale using UAV based images in order to spatially analyze yield within the field. In one of the earliest studies on this topic, Davis and Wilkinson (2006) used satellite imagery of wheat crop (visible, infrared and radar) and an ANN model showing promising results (error slightly above 10%) for a single field (≈ 36 ha). Khanal et al. (2018) employed various machine learning algorithms (including neural networks) and aircraft based multi-spectral images to predict corn yield on a single field (17.5 ha). A few studies have applied ANN's for classifying crops (Rebetez et al., 2016) and yield (Pantazi et al., 2016) at the intra-field scale. However, rather than classifying within yield categories this study aims at quantitative predictions. Models at intra-field scale would offer the individual farmer the possibility of in-season monitoring of crop, which would enable decision support systems for interventions necessary to achieve higher yields. Models trained at large regional scales rarely extrapolate to finer scales, though efforts are underway to develop scalable models (Donohue et al., 2018). The methodology introduced by You et al. (2017) shows great potential and as authors claim its scalability, it would certainly be of interest in testing at the intra-field scale.

One important aspect of remote sensing based yield prediction has been finding image channels or indices containing the most discriminating features necessary for analysis (Panda et al., 2010). Consequently, the finding in this study that the RGB images perform better than NDVI, assumes significance and aligns with the study for estimating biomass and crop height (Näsi et al., 2017). This indicates that multiple spectral bands increase the information content in comparison to the condensed NDVI image. From a utility perspective, RGB cameras are cheaper with most commercially available UAVs already fitted with decent cameras able to produce images of high resolution. Models that can perform well without the need for expensive specialized equipment will make the analyses accessible to an individual farmer.

The relationship between crop yield and its environment is non-linear and may not be sufficiently contained in the features captured by images. As shown by the studies reporting low prediction error levels, by adding multi/hyper spectral data, temporal image data, soil and environmental features in the feature matrix, it is possible to constrain the resulting model error effectively. Considering that this study models the yield based only on images, the resulting prediction error of 8.8% is promising. Additionally, collection of multi-year yield maps from sensor-equipped harvesters would add valuable information to act as ground truth. More than 90 hectares of fields were mapped in this study (2017 season). In 2018 a similar set of data has been acquired while the data acquisition will be continued in 2019. This valuable database will serve to further train, tune and verify the current model for greater accuracy. An additional limitation of this study is that only minimal preprocessing was applied to the source data. Developing automated error correction methods for data preprocessing would be another important task when developing remote sensing based crop yield models. Careful artifact rejection and preprocessing would probably benefit the modeling considerably.

In conclusion, this study is an important step towards establishing a combined model for wheat and barley yield prediction in the Finnish continental subarctic climate. The long summer growing days in this region presents a unique profile of temperature and photoperiod, justifying a region specific deep learning model for these crops. By collecting data using commercial off-the-shelf UAV and camera packages, we focus our attention on a spatial scale that enables us to predict intra-field yield distribution within the context of individual farm crop monitoring. The results indicate that the CNN models are capable of reasonable accurate yield estimates based on RGB images. It is worth noting that the CNN architecture seemed to be performing better with RGB images than NDVI images. In the future, the developed model will be trained on a larger set of features (climate and soil) along with time series image data to tune the trained model for accuracy.

Acknowledgments

We would like to give special acknowledgments to Mtech Digital Solutions Oy for partly funding this research. We also want to thank the MIKÄ DATA project's research group of Tampere University of Technology for providing the data and additional knowledge required to use the data appropriately. A special thanks to Mikko Hakojärvi from Mtech Digital Solutions Oy for providing insight into the agricultural knowledge domain.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.compag.2019.104859.

References

Bose, P., Kasabov, N.K., Bruzzone, L., Hartono, R.N., 2016. Spiking neural networks for crop yield estimation based on spatiotemporal analysis of image time series. IEEE Trans. Geosci. Remote Sens. 54 (11), 6563–6573.

Bottou, L., 1998. On-line Learning in Neural Networks. Cambridge University Press, New York, NY, USA Ch. On-line Le, pp. 9–42.

Chlingaryan, A., Sukkarieh, S., Whelan, B., 2018. Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: a review. Comput. Electron. Agric. 151 (November 2017), 61–69.

Chunjing, Y., Yueyao, Z., Yaxuan, Z., Liu, H., 2017. Application of convolutional neural network in classification of high resolution agricultural remote sensing images. ISPRS – Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-2/W7, 989–992.

Davis, I.C., Wilkinson, G.G., 2006. Crop yield prediction using multipolarization radar and multitemporal visible/infrared imagery. In: Proc.SPIE 6359, 6359–6359 – 12.

Donohue, R.J., Lawes, R.A., Mata, G., Gobbett, D., Ouzman, J., 2018. Towards a national, remote-sensing-based model for predicting field-scale crop yield. Field Crops Res. 227, 79–90.

Goodfellow, I., Bengio, Y., Courville, A., 2016. Deep Learning. MIT Press.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision 2015 Inter, pp. 1026–1034.

Hinton, G., Srivastava, N., Swersky, K., 2014. Neural Networks for Machine Learning Lecture 6a: Overview of minibatch gradient descent.

Ioffe, S., Szegedy, C., 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift.

Jiang, D., Yang, X., Clinton, N., Wang, N., 2004. An artificial neural network model for

estimating crop yields using remotely sensed information. Int. J. Remote Sens. 25 (9), 1723–1732.
Kamilaris, A., Kartakoullis, A., Prenafeta-Boldú, F.X., 2017. A review on the practice of big data analysis in agriculture. Comput. Electron. Agric. 143, 23–37.
Karpathy, A., Fei-Fei, L., 2017. Deep visual-semantic alignments for generating image descriptions. IEEE Trans. Pattern Anal. Mach. Intell. 39 (4), 664–676.
Kaul, M., Hill, R.L., Walthall, C., 2005. Artificial neural networks for corn and soybean yield prediction. Agric. Syst. 85 (1), 1–18.
Khanal, S., Fulton, J., Klopfenstein, A., Douridas, N., Shearer, S., 2018. Integration of high resolution remotely sensed data and machine learning techniques for spatial prediction of soil properties and corn yield. Comput. Electron. Agric. 153 (August), 213–225.
Krizhevsky, A., Sutskever, I., Hinton, G.E., 2017. ImageNet classification with deep convolutional neural networks. Commun. ACM 60 (6), 84–90.
Laine, A., Högnäsbacka, M., Niskanen, M., Ohralahti, K., Jauhiainen, L., Kaseva, J., Nikander, H., 2017. Virallisten lajikekokeiden tulokset 2009-2016, 262.
Matikainen, L., Karila, K., Hyyppä, J., Puttonen, E., Litkey, P., Ahokas, E., 2017. Feasibility of multispectral airborne laser scanning for land cover classification, road mapping and map updating. ISPRS - Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-3/W3, 119–122.
Milioto, A., Lottes, P., Stachniss, C., 2017. Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in CNNs. Adv. Intell. Syst. Comput. 531, 105–121.
Miyoshi, G.T., Imai, N.N., de Moraes, M.V.A., Tommaselli, A.M.G., Näsi, R., 2017. Time series of images to improve tree species classification. ISPRS – Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-3/W3, 123–128.
Näsi, R., Viljanen, N., Kaivosoja, J., Hakala, T., Pandžić, M., Markelin, L., Honkavaara, E., 2017. Assessment of various remote sensing technologies in biomass and nitrogen content estimation using an agricultural test field. ISPRS – Int. Arch. Photogramm. Remote Sens. Spatial Inf. Sci. XLII-3/W3, 137–141.
Panda, S.S., Ames, D.P., Panigrahi, S., 2010. Application of vegetation indices for agricultural crop yield prediction using neural network techniques. Remote Sens. 2 (3), 673–696.
Pantazi, X.E., Moshou, D., Alexandridis, T., Whetton, R.L., Mouazen, A.M., 2016. Wheat yield prediction using machine learning and advanced sensing techniques. Comput. Electron. Agric. 121, 57–65.
Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A., 2017. Automatic differentiation in PyTorch. In: NIPS-W.
Rebetez, J., Satizábal, H.F., Mota, M., Noll, D., Büchi, L., Wendling, M., Cannelle, B., Pérez-Uribe, A., Burgos, S., 2016. Augmenting a convolutional neural network with local histograms - a case study in crop classification from high-resolution uav imagery. In: 24th European Symposium on Artificial Neural Networks, ESANN 2016, Bruges, Belgium, April 27–29, 2016.
Ruß, G., 2009. Data mining of agricultural yield data: a comparison of regression models. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 5633 LNAI, 24–37.
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. Vision 115 (3), 211–252.
Sa, I., Chen, Z., Popovic, M., Khanna, R., Liebisch, F., Nieto, J., Siegwart, R., 2017. weedNet: Dense Semantic Weed Classification Using Multispectral Images and MAV for Smart Farming.
Tiusanen, J., 2017. Aineiston käsittely ja muotoilu. Käytännön Maamies.
Wolfert, S., Ge, L., Verdouw, C., Bogaardt, M.J., 2017. Big data in smart farming – a review. Agric. Syst. 153, 69–80.
You, J., Li, X., Low, M., Lobell, D., Ermon, S., 2017. Deep Gaussian process for crop yield prediction based on remote sensing data. In: 31th AAAI Conference on Artificial Intelligence, pp. 4559–4565.
Zeiler, M.D., 2012. ADADELTA: An Adaptive Learning Rate Method. undefined.

PUBLICATION
II

A Data Driven Approach to Decision Support in Farming
N. Narra, P. Nevavuori, P. Linna and T. Lipping

Information Modelling and Knowledge Bases XXXI 2020
DOI: 10.3233/FAIA200014

A Data Driven Approach to Decision Support in Farming

Nathaniel NARRA [a,1], Petteri NEVAVUORI [a,b], Petri LINNA [a] and Tarmo LIPPING [a]
[a] *Computing Sciences Unit, Tampere University, Finland*
[b] *Mtech Digital Solutions Oy*

Abstract. Precision Agriculture and Smart Farming are increasingly important concepts in agriculture. While the first is mainly related to crop production, the latter is more general, which also involves the carbon capture capacity of crop fields (Carbon Farming), as well as optimization of the farming costs taking into account the dynamics of market prices. In this paper we present our recent work in building a web-based decision support system for farmers to help them comply with these trends and requirements. The system is based on the Oskari platform, developed in Finland for the visualization and analysis of geospatial data. Our main focus so far has been in developing tools for Big Data and Deep Learning based modelling which will form the analytical engine of the decision support platform. We first give an overview on the various applications of deep learning in crop production. We also present our recent results on within-field crop yield prediction using a Convolutional Neural Network (CNN) model. The model is based on multispectral data acquired using UAVs during the growth season. The results indicate that both the crop yield and the prediction error have significant within-field variance, emphasizing the importance of developing field-wise modelling tools as a part of a decision support platform for farmers. Finally, we present the general architecture of the overall decision support platform currently under development.

Keywords. Smart farming, crop yield prediction, decision support, deep learning

1. Introduction

For ages, farmers have made notes on their farming activities to undertake proper actions to increase the productivity of their fields. The means and extent of these actions have changed in time - instead of digging ditches using spades, whole fields can be levelled by modern powerful machinery and fertilizers and pesticides are used to increase the yield. However, much of the decision-making regarding these modern means of cultivation is still done by intuition. At the same time, increasingly strict environmental regulations concerning farming and competition on the crop market forces farmers to optimize their cultivation activities to the limits. This optimization has multiple targets such as yield, carbon capture, environmental requirements, market prices etc. As in many other industries, data-driven modelling of production and developing model-based decision support systems has become an active area of research and development in agriculture [1].

[1]Corresponding Author: Pohjoisranta 11A, Tampere University, Pori, Finland; E-mail: nathaniel.narra@tuni.fi.

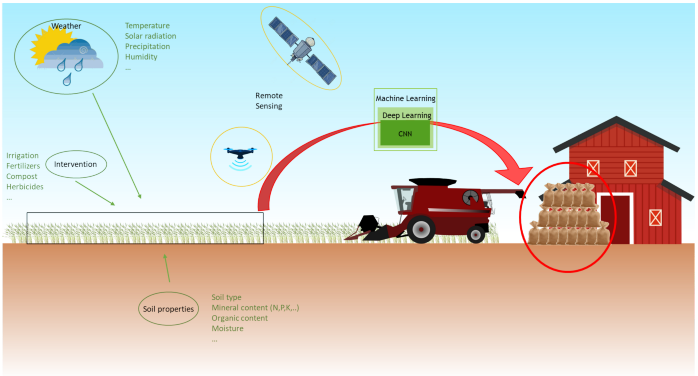

Figure 1. Illustration of crop production as a system with big data input and yield as the output.

Crop production can be viewed in engineering parlance as a system with input and output (see Figure 1). Climate, soil and other biotic and abiotic factors that have a bearing on plant growth (i.e. system dynamics) can be considered as input. These also include interventions conducted to either stimulate plant productivity or mitigate factors detrimental to productivity. With increasing accessibility in terms of affordability, ease of use and technical reliability, Internet of Things (IoT) and remote sensing technologies have enabled high amounts of data to be collected from crop fields. These data can either represent the input factors directly or constitute an indirect representation of the effects of these factors on the system (i.e. crop). Multi- or hyperspectral remote sensing is a common example of the latter type of data. Having all these data available, often in real time, opens up new avenues for studying the contribution of various factors to the yield (i.e., the output of the system).

The collected data comes in vast amounts and its analysis involves high computational cost that often preclude traditional analytical methods. Also, yearly variation in various factors such as climate, for example, suggests that the analytical tools used for decision support in agriculture have to be capable of learning from the environmental conditions. For example, in addition to training a model for crop yield prediction, it is also important to learn how the model needs to be adapted to the changes in the environment. Recent developments in Artificial Intelligence (AI) and, more specifically, in Machine Learning (ML) have produced promising new models for extracting information from large heterogeneous data sets. These methods have been extensively applied to study various aspects of agriculture [2][3][4].

A common term for the recent advancements in ML is Deep Learning (DL). DL refers to Neural Network type structures containing multiple computational layers with often thousands (or even millions) of parameters to be adapted in the training phase. Probably the most widely used deep learning structure is that of Convolutional Neural Networks (CNNs), proved to be superior in a variety of image analysis tasks. Other common structures include Long Short-Term Memory (LSTM) networks used for modelling sequences of data such as text, for example, and Generative Adversarial Networks (GANs), designed especially for generating new data based on certain features charac-

teristic to the training data set. A common property of the deep learning structures is that training of the models is performed based on data, i.e., no predefined and pre-calculated feature vector is needed. This, however, implies that extensive data sets are required for training the models and the operation principles of the models are usually not revealed.

In this study we present our recent work in designing a decision support platform for farmers. A central component of the platform is its analytical engine, involving machine learning models for various phenomena. Before presenting the general structure of the platform we present an overview on recently developed applications of deep learning in agriculture. We also present a case study on the development of crop yield prediction model using CNNs.

2. Applications of Deep Learning in Agriculture: Overview

The developments in DL algorithms, and importantly the deployment of numeric software tools to implement them, have resulted in a surge in their applications. Agriculture has been the domain of some such applications, indicated by a sharp increase in the number of publications applying DL methods to different areas of agriculture. In one of the earliest works, Kamilaris & Prenafeta-Boldú [5] review 47 published studies and recognize 16 topical areas. They further concentrate on CNN, a specific framework within DL, and review agriculture related studies using this methodology [6]. For the purpose of this study, we chose to focus on the literature considering crop production in open fields and related issues, thus excluding topics such as greenhouse farming, land-use classification, animal husbandry and fruit/orchard plantations (see Figure 2). This selection of scope was due to our ongoing work on crop yield monitoring of mainly wheat and barley fields in Finland.

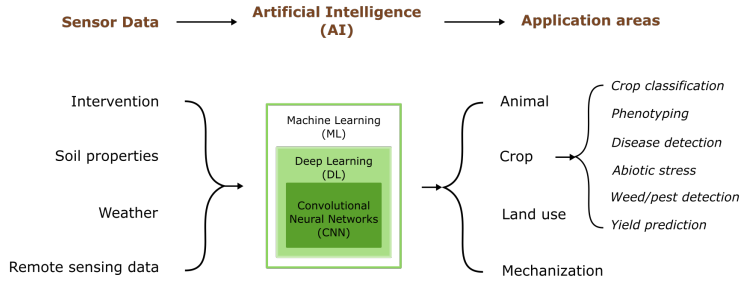

Figure 2. Application areas of DL in agriculture. In this study we focus on crop production, identifying six specific problems that can be targeted using DL models.

2.1. Crop recognition/classification

Crop recognition and classification using DL algorithms is generally relevant when the objective is to ascertain crop coverage over a large region (covering multitude of farms) based only on remote sensing images. The task can be to detect a single crop or a set of

crops. CNN based DL models have performed well in comparison to other ML methods, reaching very high classification accuracy ($> 85\%$) [7][8]. Most studies addressing this task use satellite data, but UAV imagery can also be used [9].

2.2. Phenotyping

Crop development can be assessed by quantifying the quality, structure or biomass productivity of the plants in a series of developmental stages. Ascertaining these phenological stages of plants can be important in precision agriculture for monitoring crop condition. This has implications for timing of harvest, pest control, yield prediction, farm monitoring and disaster warning. Various measures of performance can be used such as leaf counting, growth stage classification or plant maturity (age regression). Image based DL approaches have been shown to be superior to analyses based on hand crafted features [10][11]. In a recent article, Mochida et al. present an overview of various image based phenotyping studies that employ ML techniques [12].

2.3. Disease Detection

Disease, due to biotic stressors, of crops is a prime topic for testing the efficacy of DL methods in monitoring crop health. DL methods have show significant potential in improving the speed, accuracy and reliability in early detection of diseases [13]. Golhani et al. have presented an excellent review of neural network based approaches to disease detection using hyperspectral images [14]. Among the studies they review, a couple of CNN based studies performed especially well. Such studies tend to require higher resolution images and thus are most suitable for UAV based imagery. Though hyperspectral cameras are expensive currently, with falling costs they have the potential to be employed as an essential farm monitoring tool in the near future.

2.4. Abiotic Stress Detection

Abiotic stress is often unavoidable, especially in open-field cropping, and monitoring their expression in plants is important in mitigating their detrimental effect on crop productivity [13]. The stressors can be, for example, herbicide damage, water excess/deficiency, temperature extremes, nutrition deficiency. Using DL to detect and classify stress states, has resulted in superior performance in comparison to traditional regression methods.

2.5. Weed Detection

As with disease, weeds and pests can also reduce crop productivity significantly. This is essentially a task of identifying weeds and discriminating them from the crop by using detection/classification strategies. Early detection is of importance, which can be effectively accomplished using high resolution data able to capture the weeds at early stages of growth. Thus aerial and terrestrial autonomous vehicle based remote sensing systems are ideally suited for data collection. DL frameworks applied to UAV imagery have shown good results in accurately detecting and delineating specific weeds among crops [15][16]. However, this is a very challenging task and highly dependent on the specific context of the crop type and weed type. Visual similarity of the crop and weed or occlusion of the weeds in images can significantly complicate the analysis procedure.

2.6. Yield Prediction

All efforts in crop monitoring ultimately seek to improve crop productivity, i.e., yield. Earliest attempts at harnessing the potential of DL methods in predicting yield were made with encouraging results ($> 80\%$)[17]. Panda et al., used neural networks with multiple vegetation indices to predict corn yield with high accuracy ($83.5\% - 96\%$) [18]. Typically the output of the prediction model is in terms of yield classes (i.e. high, medium or low). Elavarasan et al. in their review of ML studies in yield prediction include studies with DL based yield prediction [19]. One of the interesting studies conducted by You et al (2017) used a combination of CNN and LSTM networks to predict soybean yield at regional level with very high accuracy [20]. Their method has the potential for scaling down to intra-field yield prediction.

3. Case Study: Prediction of Yield of Wheat and Barley Fields in Satakunta, Finland

DL models represent the data that they are trained on. As the growth of crops depends on climate and sunlight conditions, the variation of these conditions in time and space will potentially pose a challenge for a universal model. Thus there is a need for training models specific to regional conditions. Keeping this in mind, an effort was made to test the feasibility of using CNN models to predict wheat and barley yield grown in the Finnish continental subarctic climate.

3.1. Materials

Six fields, located in the Satakunta region of Finland near the city of Pori, were selected for this study. They vary in size, together accounting for 54.2 ha of land area. The data acquisition was conducted during the 2017 growing season. Image data were acquired using a UAV (Airinov Solo 3DR) with a multispectral camera (SEQUIOA, Parrot) mounted to it. Images were acquired in the early stage of crop growth, within 25% of the total thermal time of the respective crop variety. Pertinent details about the test fields are provided in Table 1. Crop yield data was collected in September 2017 using two sensor systems (Trimble CFX 750 and John Deere Greenstar 1) mounted to combine harvesters. Growth phase was determined by calculating the cumulative daily thermal time commencing from the date of sowing for each field. Thermal time for each day was calculated using Eq.(1), based on the daily mean temperature calculated at specific times ($t = \{02{:}00, 05{:}00, 08{:}00, 11{:}00, 14{:}00, 17{:}00, 20{:}00, 23{:}00\}$):

$$Th_t = \max\left(\left(\frac{1}{8}\sum T_t\right), 5\right) - 5 \tag{1}$$

3.2. Methods

The yield data from the harvester mounted sensors were contained in *shape* files (a file format for vector type geospatial data). The yield information is represented by polygons with an attribute describing the yield (in kg) collected over the area of the polygon.

Table 1. Details of crop fields and crop varieties in the 6 test fields. Thermal time for each crop variety is the total thermal time to crop maturity. The data to calculate the thermal time is taken from [21]. Sowing dates and imaging dates are used to calculate the growth phase as a fraction of the total thermal time for each particular crop variety.

Field #	**Size (ha)**	**Crop: (*Variety*)**	**Thermal time**	**Sowing date**	**Imaging date**	**Growth phase**
1	5.14	Barley:*Trekker*	979.7	16 May	8 Jun	15 %
2	2.97	Barley:*Trekker*	979.7	17 May	8 Jun	15 %
3	4.66	Barley:*Propino*	981.4	15 May	15 Jun	22 %
4	7.29	Barley:*Propino*	981.4	15 May	15 Jun	22 %
5	15.28	Barley:*Trekker*	979.7	18 May	1 Jun	10 %
6	18.86	Wheat:*KWS Solanus*	1065	13 May	15 Jun	21 %

These were converted to point data (polygon centroids) attributed with the yield density (kg/ha). This point data was then interpolated and rasterized to serve as the ground truth in training the DL model. The FarmWorks software tool was used in preprocessing the yield data.

The high resolution ($0.31 \times 0.31m$) images collected using the multispectral camera were compiled as mosaics using the Pix4D software tool and masked with the shape of respective fields. Two types of data sets were constituted from the measurements – 3-band RGB images and 1-band Normalized Difference Vegetation Index (NDVI) data.

A CNN model was constructed using the PyTorch [22] software library and refined through iterative tuning of relevant parameters such as: network depth (i.e., the number of convolutional layers of the CNN), the weights of the training algorithm, the hyperparameters of the training algorithm and the parameters of the regularization method. Additionally, three different image frame sizes ($10m, 20m$ and $40m$) were tested to determine the best image size to be fed to the CNN model. After all tests were performed, the best performance was observed with $40m \times 40m$ RGB image frames fed to a CNN network with 6 convolutional layers using the Adadelta training algorithm (learning rate = 0.008, past iterations' error adjustment coefficient = 0.58) with L2 regularization (weight decay = 0.001) and early stopping (patience = 50).

The CNN takes three $40m \times 40m$ image frames (1 per channel in RGB) and outputs a single density value (predicted yield). The resulting point data is georeferenced, representing the yield density predicted over the area of the image frame. In order to observe the capacity of the model to represent the spatial distribution of yield within a field, the point data was rasterized to visualize the predicted yield as a composite image of a single field.

3.3. Results

The ability of the CNN model to represent the yield distribution for each field is illustrated by the scatter plots in Figure 3. While the trend lines have similar slopes for each of the 6 fields, the data indicate a consistent pattern of overestimating low yields and underestimating high yields. In order to illustrate the prediction error relative to the magnitude of the yield, the mean absolute percentage error (MAPE) for each field is presented in (Figure 4). It can be seen that among the 6 fields the average percentage error is within 6% – 14%, with corresponding medians within 4% – 10%. The largest field (#6: 18.86 ha) was chosen to illustrate the ability of the model to follow the spatial yield

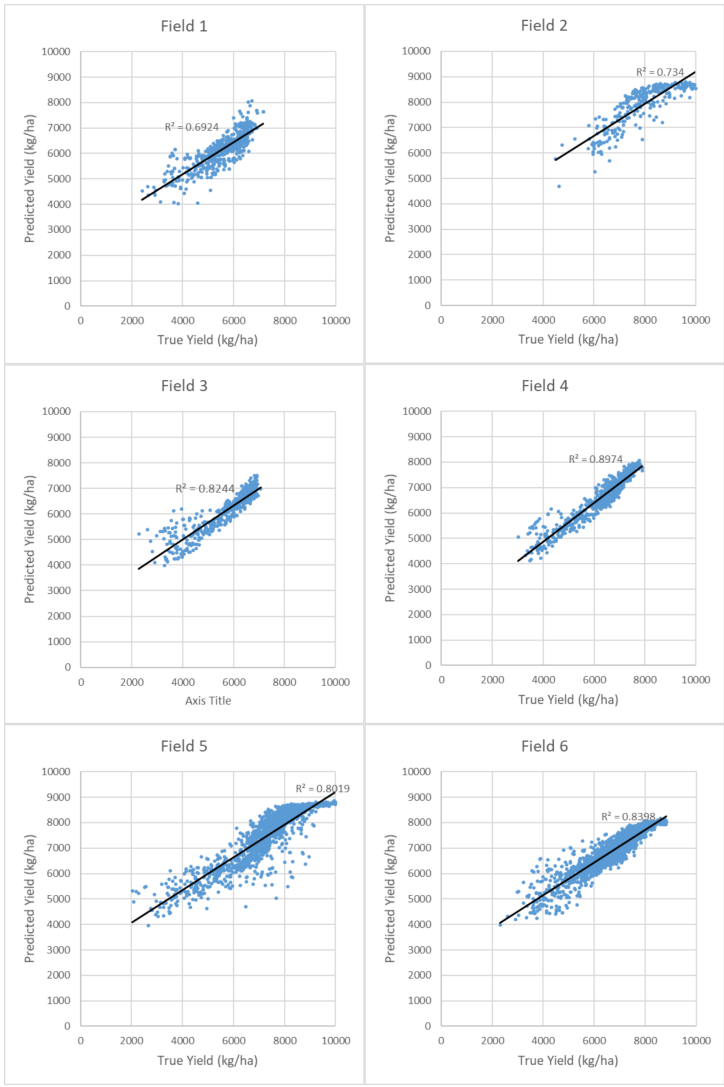

Figure 3. Correlations between the true and predicted yield for each of the 6 fields included in the study.

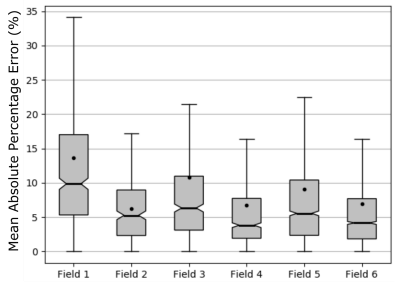

Figure 4. Boxplots of percentage error between true yield and predicted yield for each field.

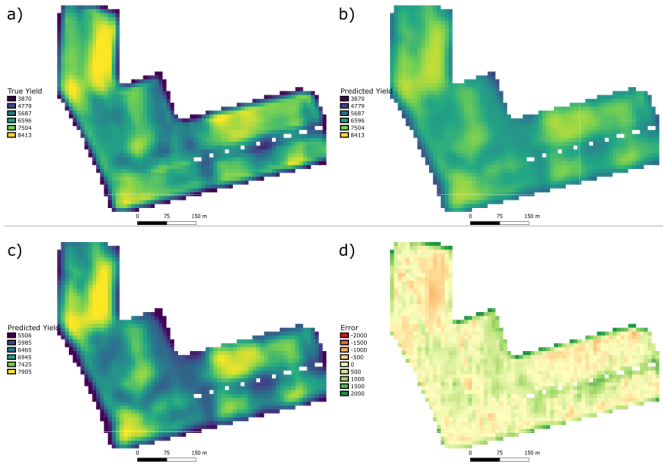

Figure 5. Visualisation of the true and predicted yield of Field 6. a) The spatial distribution of yield as recorded by the yield sensor on the combine harvester. b) Yield predicted by the CNN model. c) Predicted yield with colour-scale adjusted to min-max range. d) Error between predicted and true yield.

distribution within a field (Figure 5). The raster of the predicted yield when viewed as a colour map (Figure 5c) clearly illustrates the capability of the model to predict the spatial variations of the true yield. However, Figure 5b illustrates that the model is only capable of representing a limited range of values of the true yield. Figure 5d shows that the errors (calculated by subtracting true yield from predicted yield) are mostly in the high and low end of the range of values of true yield; thus the model over-predicts in low yield regions and under-predicts in high yield regions.

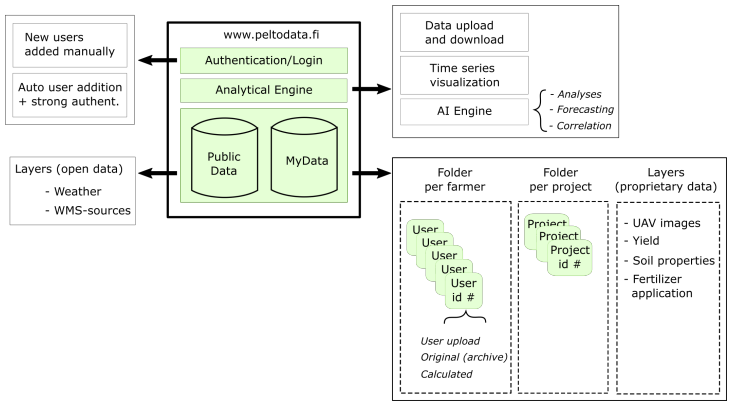

Figure 6. Structure of the Oskari based decision support platform for farmers.

4. Decision Support for Farmers: the Oskari Platform

A project was initiated with the goal of implementing a web based data repository as well as an analysis and decision support platform tailored to farmers' needs. The project explored various services and platforms and evaluated suitability based on their capability to handle access rights of farmers to their uploaded/transferred data on an individual basis. At the same time, emphasis was also placed on using open-source frameworks and contributing towards open data [23]. Consequently, mapping existing solutions revealed about 150 different platforms, though this search was not exhaustive. Overwhelming majority of the platforms were found to be either paid-for services and/or closed-source software and were therefore considered unsuitable. Among the few suitable platforms, Oskari was chosen. Oskari is an open-source (*www.oskari.org*; licence: MIT & EUPL) tool for web mapping applications using distributed spatial data infrastructure like Geoserver running as the back-end. Its front-end allows data management and custom visualisation based on an HTTP server and Java servlet extension.

The Oskari service for this project was implemented such that it can be accessed through the *peltodata.fi* domain. The architecture envisaged at this stage is illustrated in Figure 6. Through the web portal farmers can access their personal, authenticated accounts, upload data for visualization and call on AI based analytical tools for decision support. The technology to implement the analytical tools in the web based service environment has not yet been decided; the most promising options are the Shiny environment using the R language or the Python environment which has best support for DL models. The models implemented so far, including the CNN based yield prediction model, have run on a separate computer cluster.

The Data available to the farmers includes open source and proprietary data. Examples of open source data include weather data, satellite image data and land drainage maps, for example. Farm specific harvester yield maps, UAV based remote sensing image data (multispectral) and soil nutrient content maps are examples of proprietary data whose access is restricted and controlled by their owners.

5. Discussion and Conclusions

Application of ML and DL methods to agricultural (big) data has gained a lot of attention recently. The variety of problems addressed using these methods is wide, ranging from fruit counting to political decision making. In this paper we have focused on decision support for farmers cultivating open-field crops. Even in this restricted scope, there are various tasks that could be addressed by ML and DL as indicated in section 2.

As a case study, we have presented a CNN based yield prediction model, implemented and evaluated using UAV-acquired multispectral data from 6 crop fields in Satakunta, Finland. The results of the case study indicate that it can, with decent accuracy, model crop yield based on data acquired in the early phase of the growth season. Significantly, the model is capable of predicting within-field patterns of yield variation with good similarity to true yield. Training DL algorithms requires large amount of data. Kamilaris & Prenafeta-Boldú [5] lists some of the openly available datasets for training and possibly benchmarking the models. Also, training of the model and tuning of the model parameters is of high computational complexity and therefore cannot be performed on-line as a part of a decision support platform. Once trained, using the model for yield prediction is computationally relatively inexpensive. It remains to be studied how well the tuning of the algorithm can be generalized to the data from other areas and/or acquired in different years. Learning the effects of climate and other environmental conditions on the model efficiency is a long term research pursuit as data from different regions and weather conditions needs to be acquired and analyzed. Also, employing time sequences of data possibly using the LSTM DL networks would be a promising research area.

The presented case study revealed some limitations of the CNN model in yield prediction. The model underestimated/overestimated the yield in the regions of high/low yield values, respectively. The reason for this kind of behavior needs to be investigated. Another limitation is related to yield data pre-processing. In some cases the polygons of yield data overlap causing errors in yield density maps. This limitation will be addressed in the future by more careful pre-processing flow.

In its current form, the peltodata.fi portal aims to provide a few key services to the local farming community. Currently, the farmers can explore the harvester yield distribution, soil properties maps, UAV multispectral images among other open source maps. The farmers can also avail of the analyses such as predicted yield. The collaborating farmers will be involved in the development of the service to serve their needs most appropriately. With regards to the platform, Oskari has been adopted by several municipalities and government agencies in Finland, thereby forming a considerable user base. This has resulted in a core group of Oskari developers monitoring the trends and customer requirements to develop appropriate solutions. In addition, there are a lot of interesting data interfaces available, for example, from the Spatineo Director. (https://directory.spatineo.com/). An important aspect to be implemented in the future is the capability of data trading or download/port to smart devices for intervention (e.g. application of fertilizer, weedicide and irrigation).

References

[1] R. Rupnik, M. Kukar, P. Vraar, D. Koir, D. Pevec, and Z. Bosni, "AgroDSS: A decision support system for agriculture and farming," *Computers and Electronics in Agriculture*, 2018.

[2] D. I. Patrício and R. Rieder, "Computer vision and artificial intelligence in precision agriculture for grain crops: A systematic review," *Computers and Electronics in Agriculture*, vol. 153, pp. 69–81, 2018.

[3] T. U. Rehman, M. S. Mahmud, Y. K. Chang, J. Jin, and J. Shin, "Current and future applications of statistical machine learning algorithms for agricultural machine vision systems," *Computers and Electronics in Agriculture*, vol. 156, pp. 585–605, jan 2019.

[4] A. Chlingaryan, S. Sukkarieh, and B. Whelan, "Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review," *Computers and Electronics in Agriculture*, vol. 151, no. November 2017, pp. 61–69, 2018.

[5] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and Electronics in Agriculture*, vol. 147, pp. 70–90, 2018.

[6] ——, "A review of the use of convolutional neural networks in agriculture," *Journal of Agricultural Science*, no. June, pp. 1–11, 2018.

[7] M. Dyrmann, H. Karstoft, and H. S. Midtiby, "Plant species classification using deep convolutional neural network," *Biosystems Engineering*, vol. 151, pp. 72–80, nov 2016.

[8] S. Ji, C. Zhang, A. Xu, Y. Shi, and Y. Duan, "3d convolutional neural networks for crop classification with multi-temporal remote sensing images," *Remote Sensing*, vol. 10, no. 2, p. 75, jan 2018.

[9] J. Rebetez, H. F. Satizábal, M. Mota, D. Noll, L. Büchi, M. Wendling, B. Cannelle, A. Pérez-Uribe, and S. Burgos, "Augmenting a convolutional neural network with local histograms - a case study in crop classification from high-resolution uav imagery," in *24th European Symposium on Artificial Neural Networks, ESANN 2016, Bruges, Belgium, April 27-29, 2016*, 2016.

[10] H. Yalcin, "Plant phenology recognition using deep learning: Deep-pheno," *2017 6th International Conference on Agro-Geoinformatics*, pp. 1–5, 2017.

[11] J. R. Ubbens and I. Stavness, "Deep plant phenomics: A deep learning platform for complex plant phenotyping tasks," *Frontiers in Plant Science*, vol. 8, p. 1190, 2017.

[12] K. Mochida, S. Koda, K. Inoue, T. Hirayama, S. Tanaka, R. Nishii, and F. Melgani, "Computer vision-based phenotyping for improvement of plant productivity: a machine learning perspective," *GigaScience*, vol. 8, no. 1, jan 2019.

[13] A. K. Singh, B. Ganapathysubramanian, S. Sarkar, and A. Singh, "Deep learning for plant stress phenotyping: Trends and future perspectives," *Trends in Plant Science*, vol. 23, no. 10, pp. 883–898, oct 2018.

[14] K. Golhani, S. K. Balasundram, G. Vadamalai, and B. Pradhan, "A review of neural networks in plant disease detection using hyperspectral data," *Information Processing in Agriculture*, vol. 5, no. 3, pp. 354–371, sep 2018.

[15] M. D. Bah, A. Hafiane, and R. Canals, "Deep learning with unsupervised data labeling for weed detection in line crops in UAV images," *Remote Sensing*, vol. 10, no. 11, 2018.

[16] H. Huang, J. Deng, Y. Lan, A. Yang, X. Deng, and L. Zhang, "A fully convolutional network for weed mapping of unmanned aerial vehicle (UAV) imagery," *PLOS ONE*, vol. 13, no. 4, pp. 1–19, 04 2018.

[17] I. C. Davis and G. G. Wilkinson, "Crop yield prediction using multipolarization radar and multitemporal visible/infrared imagery," *Proc.SPIE*, vol. 6359, pp. 6359 – 6359 – 12, 2006.

[18] S. S. Panda, D. P. Ames, and S. Panigrahi, "Application of vegetation indices for agricultural crop yield prediction using neural network techniques," *Remote Sensing*, vol. 2, no. 3, pp. 673–696, 2010.

[19] D. Elavarasan, D. R. Vincent, V. Sharma, A. Y. Zomaya, and K. Srinivasan, "Forecasting yield by integrating agrarian factors and machine learning models: A survey," *Computers and Electronics in Agriculture*, vol. 155, pp. 257–282, dec 2018.

[20] J. You, X. Li, M. Low, D. Lobell, and S. Ermon, "Deep Gaussian Process for Crop Yield Prediction Based on Remote Sensing Data," *31th AAAI Conference on Artificial Intelligence*, pp. 4559–4565, 2017.

[21] A. Laine, M. Högnäsbacka, M. Niskanen, K. Ohralahti, L. Jauhiainen, J. Kaseva, and H. Nikander, "Virallisten lajikekokeiden tulokset 2009-2016," p. 262, 2017.

[22] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, "Automatic differentiation in PyTorch," in *NIPS-W*, 2017.

[23] P. Linna, T. Mkinen, and K. Yrjnkoski, "Open data based value networks: Finnish examples of public events and agriculture," in *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2017, pp. 1448–1453.

PUBLICATION III

Assessment of Cloud Cover in Sentinel-2 Data Using Random Forest Classifier

P. Nevavuori, T. Lipping, N. Narra and P. Linna

*International Geoscience and Remote Sensing Symposium (IGARSS)*2020. Accepted for publication

ASSESSMENT OF CLOUD COVER IN SENTINEL-2 DATA USING RANDOM FOREST CLASSIFIER

P. Nevavuori

Mtech Digital Solutions Oy
Vantaa, Finland

T. Lipping, N. Narra, P. Linna

Tampere University
Tampere, Finland

ABSTRACT

In this paper, a novel cloud coverage assessment method for the Sentinel-2 data is presented. The method is based on the Random Forest classifier and the target values used in the training process are obtained by comparing the NDVI indexes calculated from the satellite and the UAV data. The developed method is shown to outperform the Sentinel Cloud Probability Mask (CLDPRB) and Scene Classification (SCL) data layers in detecting cloudy areas.

***Index Terms*—** Cloud coverage, Random Forest classifier, Crop monitoring

1. INTRODUCTION

Data from the Sentinel satellites are intensively used for various applications such as land use and vegetation mapping or crop monitoring, for example. Depending on climate conditions in the region of interest, the main obstacle in using the data for practical monitoring purposes may be cloud coverage. This is especially restricting if the data should be acquired from a narrow time window corresponding, for example, to a certain growth phase of crops. The problem could be alleviated by more accurate and higher resolution cloud coverage assessment compared to that available by the product of the Sentinel data.

Currently the cloud mask of the Sentinel data is available in the form of the Level 1C product containing vector layers of dense and cirrus clouds. Also, the percentage of cloudy pixels (dense and cirrus) in the mask are provided. The Level 2A product further processes the Level 1C data to obtain the Scene Classification layer with cloud and cirrus probability values at 60 m spatial resolution. Calouzzi et.al. [1] assessed these products concluding that caution has to be taken when using the provided cloud masks and improved cloud detection algorithms are welcome. Recently, Baetens et.al. [2] compared three cloud mask calculation algorithms: MAJA (used in the Level 2A product), Sen2Cor (used by ESA) and FMask (used by USGS), using their Active Learning Cloud Detection (ALCD) method for producing reference cloud masks. Classification accuracy of about 90 % was obtained by MAJA and FMask while SenCor gave 84 % accuracy.

In this paper we train the random forest classifier to assess cloud cover in Sentinel-2 data. Our primary usage of the data is crop monitoring and yield prediction for decision support for farmers. Therefore, the classifier is trained using data acquired from crop fields by UAVs: as UAVs fly below the clouds and the data they produce is not affected by cloud cover (if properly corrected for changes in irradiance), the difference between the UAV and Sentinel data can be used as ground truth for cloud cover.

2. DATA

2.1. Drone Images

For cloudless multispectral ground truth data, ten crop fields were selected for imaging in the vicinity of Pori, Finland (61°29'N, 21°48'E) and were imaged as a part of the MIKA DATA project [3, 4]. The total area of the selected fields was approximately 93 ha. Half of the fields had wheat (*Zebra/Mistral*), three had barley (*Harbringer/RGT Planet*) and two remaining had oats (*Ringsaker*) as the cultivated crop. The fields were imaged during the growing season for years 2018 and 2019 from the time of sowing to the time of harvest. All fields were imaged weekly. Due to varying weather conditions and the proximity of an airport, the temporal allocation of imaging flights to within a fixed daily time range was not possible. The images were thus taken during day time.

The fields were imaged with two distinct drones, using 3DR Solo for the year 2018 and Parrot Disco-Pro AG for 2019. The drones were equipped with similar Parrot Sequioa multispectral cameras. Distinct images were collated for each field to build a complete image of a field using the Pix4D software. During the process of building the image mosaics, the band data were also automatically normalized in terms of radiance utilizing the information provided by the multispectral camera's irradiance sensor. Using the red and near-infrared (NIR) channels, the normalized difference vegetation index (NDVI) was then calculated from each field's multi-band mosaic. To use the drone data in conjunction with the Sentinel-2 data, the collated drone images were downsampled to match the highest resolution available in Sentinel-2 images, 10 m/px. The downsampling was done using `cubicspline` interpolation algorithm in the `gdalwarp` utility. Lastly, the images for each field were cut to proper shape with field block border data provided by Ruokavirasto (*Finnish Food Authority*) [5]. This resulted in a total of 288 distinct crop field images. The field-wise sizes, crop varieties, yearly image counts and average valid pixel counts per image are given in Table 1

The use of NDVI images calculated from drone data is discussed in Sec. 2.3. Next we will discuss the acquisition and processing of the Sentinel-2 satellite data.

2.2. Sentinel-2 Data

Sentinel-2 satellite images were selected as the source data for the study. The data provided by the dual satellite system are widely used in agriculture and is freely available. The satellite images processed to the Sentinel product Level-2A [6] were downloaded from Copernicus Open Access Hub [7]. The satellite data products were downloaded for the growing seasons of 2018 and 2019.

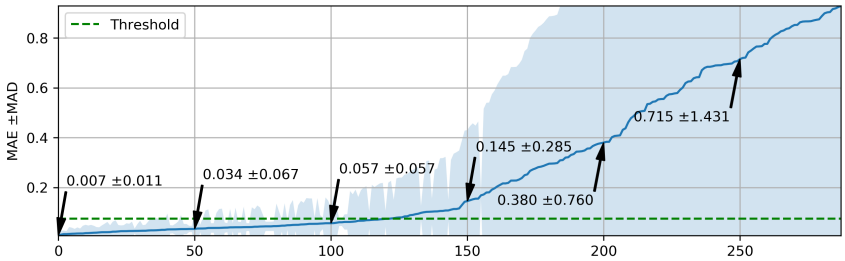

Fig. 1. The mean absolute errors (MAE) and mean absolute deviations (MAD) of week-aligned NDVI pairs in ascending order. The statistics are calculated over the pixels in the paired Sentinel-2 and drone NDVI images.

Table 1. Sizes, crops, image counts and average pixel counts of fields selected for drone imaging.

Field	Size, ha	Crop	Image Counts 2018	Image Counts 2019	Avg. Valid Px Per Image
1	11.08	Wheat	13	16	1065.5
2	8.24	Wheat	15	14	759.1
3	11.77	Wheat	13	16	1120.9
4	11.12	Wheat	15	16	1051.9
5	7.59	Wheat	15	16	705.2
6	7.61	Oats	12	15	739.8
7	7.24	Oats	13	15	681.9
8	7.77	Barley	13	15	1016.6
9	13.05	Barley	12	16	1251.3
10	7.95	Barley	12	16	715.5

The satellite data were selected with no limits on the estimated cloud coverage. The goal was to be able to find week-matching pairs for the drone data. The data was used as the training data for which information about the cloudless ground truth was available via drone data. The gathered data spanned initially the growing seasons of years 2018 and 2019. Part of the downloaded data was omitted during the process of week-matching Sentinel-2 data to Drone data. The satellite image data were cut to shape using field block borders already utilized with the drone data to ultimately generate image pairs of drone and satellite data aligned both temporally and geographically for distinct fields.

2.3. Target Data

Supervised machine learning requires the existence of *a priori* labeled data, the ground truth. With the aim of estimating cloud coverage in Sentinel-2 data in the spatial scale of crop fields, NDVI images gathered with drones at the altitudes well below clouds are considered as cloudless ground truth. This consideration is in relation to satellites flying at atmospheric altitudes. Comparing absolute values across bands for two different sensors and imaging platforms has proven to be difficult, as the data would require scaling to an unkown global maximum for Sentinel-2. However, the use of NDVI alleviates this problem by providing normalized and thus comparable data between distinct imaging systems.

Target data needs thus to be generated using the week-aligned NDVI data from both sources, the drone and the Sentinel-2 systems. Each spatially and temporally aligned satellite and drone NDVI image pair is compared pixel by pixel to determine whether the images are similar on the level of distinct pixels. A pixel correponds to an area of 10 $\times$ 10 meters. The similarity for a single pixel-corresponding area is determined by

$$sim_{(s,d)} = \begin{cases} 1, & |s-d| \leq threshold \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where s and d are spatially and temporally aligned pixels for a field from the satellite and drone sources respectively. The mean absolute errors (MAEs) of all week-aligned image pairs are depicted in Fig. 1. The determination of the threshold is discussed next.

To determine a proper absolute NDVI difference threshold for labeling Sentinel-2 pixels either similar or dissimilar to the drone pixels (see Eq. 1), the two data sources were compared using the Student t-test. The test was applied over the pixels in the images to compare whether the NDVI values in the images were statistically similar or not. A total of 15 statistically similar ($p = 0.01$) week-aligned image pairs were found. It is to be noted though, that the number of image pairs having MAE in close proximity to the similarity threshold was higher than just 15 (see Fig. 1).

The statistically similar data (15 image pairs) were then used to empirically determine the proper threshold for classifying NDVI differences in terms of pixel-wise similarity. The tested thresholds were selected from the proximity of upper end of the MAE for the statistically similar data samples as shown in Table 2.

Table 2. NDVI difference metrics for similar image pairs

Image pairs	15
Avg. Diff.	0.001 ±0.046
MAE	0.026 ±0.022
MSE	0.003 ±0.010
RMSE	0.046 ±0.092

In more general terms, the task of determining the threshold for labeling is a task of balancing between (1) capturing as much similarities while (2) still excluding as many dissimilarities as possible. To elaborate, labeling every pixel in the statistically similar images as similar would require increasing the absolute NDVI threshold to levels possibly having some pixels incorrectly labeled as similar. The ratios of pixels labelled as similar for each similar image pair with different thresholds is given in Table 3. In combination with visual

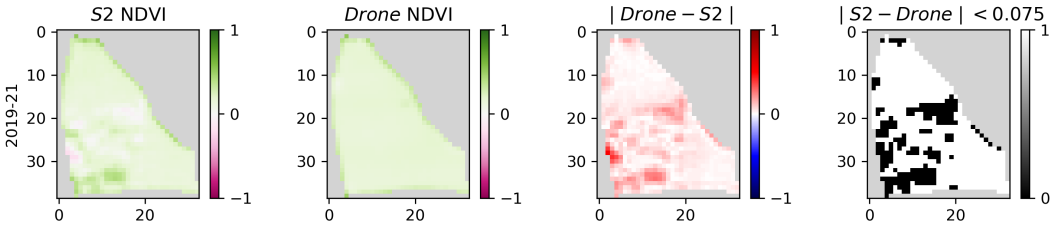

Fig. 2. A visualization of a single week-aligned Sentinel-2 and drone NDVI image pair with the absolute difference and the similarity map. The first two figures depict the NDVI maps from corresponding sources. The third figure shows the absolute difference between the aligned Sentinel-2 and drone NDVI values. The fourth figure shows the thresholded absolute difference, indicating areas where the NDVI images are similar enough.

evaluation, a threshold of 0.075 absolute NDVI difference was selected. A single image pair with the calculated similarity map is shown in Fig. 2.

Table 3. Similarity ratios with various thresholds

Threshold	Similarity
0.025	89.13%
0.050	94.40%
0.075	96.14%
0.100	97.13%

2.4. Building the Modeling Data Sets

After the generation of field and week specific similarity label maps, the data required only minor preprocessing. As the Sentinel-2 data products are delivered as separate files for distinct bands and layers, the satellite data were merged to construct multi-band images instead of multiple images of distinct bands. The following Sentinel-2 data were merged:

- *Sensor bands*: 1 to 8, 8A, 9, 11 and 12
- *Level-2A layers*: AOT, SCL, TCI, WVP and CLDPRB

The separately calculated NDVI data were also merged in conjunction with the alpha-channel generated during the processing of the data. As per machine learning best-pactices, the categorical values from the scene classification layer (SCL) needed to be separated to distinct binary raster layers according to the SCL classification labels, which is also known as transforming a multi-class representation to class-wise one-hot representation [8].

Thus, the final processed input data constituted 30 distinct layers of data for each pixel. The dataset was then created by extracting multi-band Sentinel-2 pixels as input samples and their spatially and temporally corresponding binary similarity label map pixels as target values. In other words, a single input sample was a $[1 \times 30]$ and its corresponding target sample a binary-valued $[1 \times 1]$ vector. A total of 381972 input-target samples (pixels) were extracted from the source data. The samples were then shuffled and split into training and test data sets with 190986 and 63661 samples, correspondingly. No scaling was applied due to the selected decision tree based model.

3. MODEL

Data based modeling with machine learning methods is in practice a tradeoff between model explainability and increased performance. While training an accurate model for classifying distinct Sentinel-2 pixels as similar or dissimilar to the cloudless ground truth data from drones is the primary goal while the explainability was deemed as an important objective to pursue as well. This is why an ensemble model called Random Forest from the decision tree algorithm family was selected. The ensemble model is able to model non-linear relationships, work with unscaled data and provide easily understandable explanations of decisions' causes [9]. The model implementation was part of the Python's `scikit-learn` framework [10].

Table 4. The confusion matrix of similarity label predictions.

Pred/True	**0**	**1**
0	TP 23237	FP 2580
1	FN 1807	TN 36037

4. RESULTS

The model was allowed to train 500 sub-trees, varying the tree structure and features used for each tree, using the training data set only. The performance of the model was then evaluated with the hold out test data set. The confusion matrix of model predictions against true labels is shown in Table 4. The precision of the model is

$$PPV = \frac{TP}{TP + FP} = 0.900, \tag{2}$$

where PPV stands for positive prediction value. The model's true positive rate, i.e., recall, is then

$$TPR = \frac{TP}{TP + FN} = 0.923. \tag{3}$$

The F_1-score, a statistical test accuracy measure for binary classification analysis is then calculated using Eqs. 2 and 3 by

$$F_1 = 2 * \frac{PPV * TPR}{PPV + TPR} = 0.911. \tag{4}$$

Another interesting metric is the negative prediction value

$$NPV = \frac{TN}{TN + FN} = 0.952, \tag{5}$$

which shows the model's precision in predicting dissimilarities. In conjunction with test data set result analysis, the model was also evaluated with distinct images from the original source data.

Due to Sentinel-2 satellite data being sensitive to changes and disturbances in atmospehric conditions, the cloud estimation information from the scene classification layer (SCL) and cloud probablity mask (CLDPRB) calculated in the Level-2A processing of the Sentinel-2 data can not be taken as definitive truth. They, however, form a proper baseline to which compare the trained model's performance against.

The model predictions are based on the similarities of Sentinel-2 and drone NDVI images, i.e. label 1 indicates predicted similarity. Taking a mean of a set of predicted values describes the mean predicted similarity for that set. The two cloudiness estimation masks in the Sentinel-2 data product are formulated differently.

As the name indicates, the CLDPRB mask contains pixel-wise probability values for the estimated degree of cloud coverage. The model-equivalent similarity measure would thus be

$$\text{CLDPRB}_{\text{SIM}} = 1 - \text{CLDPRB}, \tag{6}$$

where larger values imply increased degree of estimated similarity.

On the other hand, the SCL layer contains pixel-wise labels, with some labels indicating cloudiness (see [6]). To gain information about the SCL layer's model-equivalent similarity measure, the cloud-related label ratio

$$p_{cl} = \frac{\text{count}(\text{SCL}_{cl})}{\text{count}(\text{SCL})} \tag{7}$$

is first counted with the cl being a set of cloud-related class labels. The inverse

$$\text{SCL}_{\text{SIM}} = 1 - p_{cl} \tag{8}$$

can then be seen as the implied cloudless ratio for a set of samples. The comparison of sample-wise similarity estimations between the trained model and Sentinel-2 data products are given in Table 5. The estimates are given both for when the true target value was 0 (satellite differed from drone) and when it was 1 (satellite similar to drone).

Table 5. Similarity estimates with hold out test data.

	$y = 0$			$y = 1$		
	Mean	Std	Median	Mean	Std	Median
Model	**0.07**	0.25	0.00	0.93	0.26	1.00
$\text{CLDPRB}_{\text{SIM}}$	0.45	0.45	0.26	**0.97**	0.14	1.00
SCL_{SIM}	0.28	0.45	0.00	0.95	0.22	1.00
Samples		38617			25044	

5. DISCUSSION AND CONCLUSIONS

Our study indicates that the Random Forest model outperforms the Sentinel-2 CLDPRB and SCL data layers in detecting cloudy areas ($y = 0$). For non-cloudy areas the detection accuracy was slightly higher for the Sentinel products (see Table 5). Several issues should be considered, however, when comparing these results. Firstly, when training the Random Forest classifier, the thresholded absolute difference between the Sentinel-2 and drone data was used as the ground truth. While it can be argued that the main cause of this difference is cloudiness, there may also be other factors involved such as shadows or differences in irradiance. The satellite and drone imagery were not necessarily acquired during the same time of the day or same day of the week, although best time-matching pairs were looked for when selecting the data. In some cases a couple of days may cause significant changes in the crop development. Another limitation comes from using the NDVI data layers for ground truth assessment. While the NDVI index contains significant information for vegetation monitoring and is probably a good choice when assessing cloud cover in crop fields, its use reduces the generalizability of the results to other land cover types.

Despite the mentioned limitations, the developed method was found to improve the usability of Sentinel data in crop monitoring. By visual inspection it was observed that in many cases when the Sentinel-2 products indicated the whole crop field to be cloud-covered, there were still significant areas of almost clear skies. The proposed algorithm proved capable in detecting these areas with considerable accuracy.

6. REFERENCES

[1] Rosa Coluzzi, Vito Imbrenda, Lanfredi Maria, and Simoniello Tiziana, "A first assessment of the sentinel-2 level 1-c cloud mask product to support informed surface analyses," *Remote Sensing of Environment*, vol. 217, pp. 426–443, 09 2018.

[2] Louis Baetens, Camille Desjardins, and Olivier Hagolle, "Validation of copernicus sentinel-2 cloud masks obtained from maja, sen2cor, and fmask processors using reference cloud masks generated with a supervised active learning procedure," *Remote Sensing*, vol. 11, 02 2019.

[3] Petteri Nevavuori, Nathaniel Narra, and Tarmo Lipping, "Crop yield prediction with deep convolutional neural networks," *Computers and Electronics in Agriculture*, vol. 163, no. June, pp. 104859, 2019.

[4] Nathaniel Narra, Petteri Nevavuori, Petri Linna, and Tarmo Lipping, "A Data Driven Approach to Decision Support in Farming," in *Information Modelling and Knowledge Bases XXXI*, Ajantha Dahanayake, Janne Huiskonen, Yasushi Kiyoki, Bernhard Thalheim, Hannu Jaakkola, and Naofumi Yoshida, Eds., vol. 321, pp. 175 – 185. IOS Press, 2020.

[5] Ruokavirasto, "Peltolohkorekisteri," .

[6] ESA, "Level-2A Algorithm - Sentinel-2 MSI Technical Guide - Sentinel Online," .

[7] ESA, "Open Access Hub," .

[8] Danfeng Hong, Naoto Yokoya, Nan Ge, Jocelyn Chanussot, and Xiao Xiang Zhu, "Learnable manifold alignment (LeMA): A semi-supervised cross-modality learning framework for land cover and land use classification," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 147, pp. 193–205, jan 2019.

[9] Peter Flach, "Machine Learning: The Art and Science of Algorithms that Make Sense of Data," p. 409, 2012.

[10] Lars Buitinck, Gilles Louppe, Mathieu Blondel, Fabian Pedregosa, Andreas Mueller, Olivier Grisel, Vlad Niculae, Peter Prettenhofer, Alexandre Gramfort, Jaques Grobler, Robert Layton, Jake Vanderplas, Arnaud Joly, Brian Holt, and Gaël Varoquaux, "API design for machine learning software: experiences from the scikit-learn project," 2013.

PUBLICATION
IV

Crop Yield Prediction Using Multitemporal UAV Data and Spatio-Temporal Deep Learning Models

P. Nevavuori, N. Narra, P. Linna and T. Lipping

Remote Sensing 12.23 (2020)
DOI: 10.3390/rs12234000

remote sensing

MDPI

Article

Crop Yield Prediction Using Multitemporal UAV Data and Spatio-Temporal Deep Learning Models

Petteri Nevavuori [1,*], Nathaniel Narra [2], Petri Linna [2] and Tarmo Lipping [2]

[1] Mtech Digital Solutions Oy, 01301 Vantaa, Finland
[2] Faculty of Information Technology and Communication Sciences, Tampere University, 33014 Tampere, Finland; nathaniel.narra@tuni.fi (N.N.); petri.linna@tuni.fi (P.L.); tarmo.lipping@tuni.fi (T.L.)
* Correspondence: petteri.nevavuori@mtech.fi

Received: 4 November 2020; Accepted: 4 December 2020; Published: 7 December 2020

Abstract: Unmanned aerial vehicle (UAV) based remote sensing is gaining momentum worldwide in a variety of agricultural and environmental monitoring and modelling applications. At the same time, the increasing availability of yield monitoring devices in harvesters enables input-target mapping of in-season RGB and crop yield data in a resolution otherwise unattainable by openly availabe satellite sensor systems. Using time series UAV RGB and weather data collected from nine crop fields in Pori, Finland, we evaluated the feasibility of spatio-temporal deep learning architectures in crop yield time series modelling and prediction with RGB time series data. Using Convolutional Neural Networks (CNN) and Long-Short Term Memory (LSTM) networks as spatial and temporal base architectures, we developed and trained CNN-LSTM, convolutional LSTM and 3D-CNN architectures with full 15 week image frame sequences from the whole growing season of 2018. The best performing architecture, the 3D-CNN, was then evaluated with several shorter frame sequence configurations from the beginning of the season. With 3D-CNN, we were able to achieve 218.9 kg/ha mean absolute error (MAE) and 5.51% mean absolute percentage error (MAPE) performance with full length sequences. The best shorter length sequence performance with the same model was 292.8 kg/ha MAE and 7.17% MAPE with four weekly frames from the beginning of the season.

Keywords: crop yield prediction; UAV; spatio-temporal modelling; time series; deep learning; cnn-lstm; convolutional lstm; 3d-cnn

1. Introduction

The abundance of modern sensor and communication technology already present in production facilities and similar highly connected environments has also seeped into the realm of agriculture. Various globally, nationally and locally available data generating remote sensing systems are in place, providing relevant data for optimizing several agricultural outputs. On the global and national scale, satellite systems (Sentinel and Landsat missions, for example) provide temporally relevant spatial data about visible land surfaces. Nationally, there are various instruments in place to both track and predict climatological variables. Data for fields and relevant other entities is also gathered on a per-field basis by agricultural expert institutions. While satellite data is meaningful when monitoring large fields, smaller fields common to European countries, as an example, require higher resolution data. Human-operated unmanned aerial vehicles (UAV) play a key role in high resolution remote sensing in fields, that otherwise would wholly be covered by just tens or, at most, a couple hundreds of open-access satellite spatial data resolution pixels (10 $\times$ 10 m/px for Sentinel-2, for example). Also, utilizing modern sensors and global navigation satellite system (GNSS) tracking with agricultural machinery further adds detail to the pool of generated data. Modern data-based modeling techniques

also benefit from increased resolution of spatial data, as they are able to better learn the relevant features in performing a given task, e.g., intra-field yield prediction. Feeding this data to automated processing and decision making pipelines is a vital part of Smart Farming enabling Decision Support Systems [1].

In [2] we performed crop yield estimation with point-in-time spatial data, point-in-time estimation being contrary to time series regression. In this study we examined the effect of time, as an additional feature, on intra-field yield prediction. Especially, we focused on the capabilities of deep learning time series models utilizing UAV remote sensing time series data as their inputs. Firstly, we wanted to see if we could surpass the performance of the point-in-time model [2] by using spatio-temporal deep learning model architectures. Secondly, we wanted to see which spatio-temporal architecture would perform better in the same task. Lastly, we perform comparative evaluation of different sequence configurations to perform actionable crop yield predictions with data collected at the beginning of the growing season.

We utilize the properties of Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks to perform spatio-temporal modelling. The CNN is briefly discussed in Section 2.3.1 and the LSTM in Section 2.3.2. The model architectures that we implemented are the following:

CNN-LSTM. CNN and LSTM networks can be utilized as separate but sequentially connected feature extractors, where the CNN first ingests spatial data and then provides extracted spatial feature data to the LSTM [3]. These models are hereafter referred to as CNN-LSTM are discussed in Section 2.3.3.

ConvLSTM. Convolutional learning properties can also be utilized differently. Models that utilize convolutional layers embedded into the LSTM architecture in a manner that eliminates the necessity to use spatial feature extraction prior to feeding the data to a sequential model are hereafter referred to as ConvLSTMs [4] and discussed in Section 2.3.4.

3D-CNN. A fully convolutional architecture can also be used to model sequential data. It is done by applying the convolution in the direction of time (or depth) in addition to width and height dimensions of spatial data [5]. Fully convolutional models utilizing the third dimensions for convolution are hereafter referred to as 3D-CNNs and discussed in Section 2.3.5.

1.1. Related Work

Regarding data similar or related to our study, recent crop-related studies utilize satellite-based data at scales larger than single fields. Ref. [6] performed county-scale soybean yield prediction with a CNN-LSTM architecture in parts of US. In addition to US national weather and yield data, they used time series satellite data from the MODIS satellite system. The data resolution was from 500×500 m/px to 1×1 km/px. Ref. [7] performed crop type segmentation of small holder farms in Germany, Ghana and South Sudan using data from Sentinel S1, S2 (10×10 m/px) and PlanetScope (3×3 m/px) satellite systems and time of year as an additional feature. Ref. [8] performed crop type mapping with a 30×30 m/px crop-specific annual land cover data combined from various satellite data sources for the area of Nebraska, US. Ref. [9] classified crop varieties from satellite time series data frames collected by the Chinese Gaofen missions with data resolutions from 4×4 m/px up to 15×15 m/px.

In the broader context of time series modelling with remote sensing data, several recent studies utilize spatio-temporal model architectures. The US county-scale soybean yield prediction by [6] was performed using a CNN-LSTM composite architecture, where a sequence of input frames was transformed into vectors of spatial features and then fed to an LSTM. Ref. [7] employed both a CNN-LSTM and a 3D-CNN model to perform crop type classification in Ghana and South Sudan, feeding multi-layer remote sensing time series data frames to the models. Ref. [8] built and trained a bidirectional ConvLSTM to predict crop maps from satellite data at the early stages of the growing season in Nebraska. While their main contribution was to affirm the feasibility of such model, they also

employed their model in a CNN-LSTM setting, using pre-trained CNN called VGG11 [10] to extract spatial features from sequences of past crop map images and then feeding these sets of spatial-like features further to the ConvLSTM. Ref. [9] used a 3D-CNN architecture in their crop type mapping study, feeding sequences of RGB image data from distinct areas to the network thus having the model learn both spatial and temporal features from the data. Ref. [11] built and trained a bidirectional ConvLSTM to automatically extract meaningful features from hyperspectral data consisting of several hundred bands for land cover pixel classification. They utilized the sequential modeling power of the ConvLSTM for feature extraction from individual images, feeding distinct bands to the model as if they were items in a sequence. Among other models, they also trained a 3D-CNN for the task to compare performance. Ref. [12] utilized a Gated Recurrent Unit (GRU) in building the convolutional recurrent model, i.e., ConvLSTM-like architecture. Their domain of application was in utilizing novel machine learning methodologies in performing land cover classification from satellite data. They employed their ConvLSTM-like model in parallel with a CNN to produce pixel-level land cover classification and report improved performance against widely utilized decision tree models for similar task. Ref. [13] employed the ConvLSTM in a encoder-decoder architecture to predict maize growth stage progression using several meteorological features using nationally collected meteorological data in China. The ConvLSTM was used as a feature extracting encoder while the decoder was an LSTM producing a desired output sequence. They modified the ConvLSTM to perform 1D convolutions on row-like data. A CNN-LSTM was also trained for comparative purposes. Ref. [14] compared the performance of 3D-CNNs against other deep learning architectures in the task of performing scene classification based on hyperspectral images. While the domain of their application is that of spectral-spatial and not spatio-temporal, they report 3D-CNN performing the best among other tested model compositions. Ref. [15] employ a wide array of CNN configurations to perform yield estimation using soil and nutrient information available pre-season arranged as spatial data. Most relevant to our study is their utilization of a 3D-CNN architecture which they use to ingest point-in-time data and learn salient features across varying input data rasters to estimate the crop yield.

1.2. Contribution

In contrast to studies performed at larger spatial scales, the main contribution of our study is to perform time series based intra-field yield prediction with multi-temporal data collected during the growing season with UAVs. In the context of using of remote sensing data in performing data-based modeling to aid in Smart Farming, we perform time series regression with remote sensing data, which is both collectable using commercially available UAVs and has spatial resolutions well below 1×1 m/px. We also use meteorological information, cumulative temperatures, to inform the models about change between weekly data. Our study builds on [2], introducing an extra variable to the modeling task, time, to see whether using time series data is more beneficial than using point-in-time data only. We develop, train and compare several spatio-temporal models to determine the most suitable model for intra-field yield modelling from a selection of models already utilized in the context of spatio-temporal modelling with remote sensing data. To also see if the spatio-temporal models can be used with a limited sequence of data from the beginning of the growing season, we evaluate the predictive capabilities and, thus, the usability, of the best performing model by feeding it time series data limited in this manner.

2. Materials and Methods

2.1. Data Acquisition

RGB images. Nine crop fields totaling to approximately 85 ha and having wheat, barley and oats as the crop varieties, were included in the study. The data was acquired during the year 2018 in the proximity of Pori, Finland (61°29′6.5″ N, 21°47′50.7″ E). Specific information about the fields is given in Table 1. The fields were imaged with a SEQUIOA (Parrot Drone SAS, Paris, France) multispectral

camera mounted on a Airinov Solo 3DR (Parrot Drone SAS, Paris, France) UAV from the average height fo 150 m using a minimum of three ground control points for each field and preflight color calibration. The imaging was done weekly, from week 21 to 35 and spanning 15 weeks in total. Due to weather conditions precluding UAV flight, gaps in data were present. For each field-specific set of images, a complete mosaic image of a field was constructed with Pix4D (Pix4D S.A., Prilly, Switzerland) software and cut to match the shape of the field boundaries. Radiometric correction was perofrmed using the illumination sensor of the Sequioa camera. The field image data was used as inputs to perform predictions with the considered models.

Table 1. The fields selected for the study in the proximity of Pori, Finland. The thermal time is calculated as the cumulative sum of temperature between the sowing and harvest dates. Mean yield has been calculated from processed yield sensor data for each field.

Field Number	Size (ha)	Mean Yield (kg/ha)	Crop (*Variety*)	Thermal Time	Sowing Date
1	11.11	4349.1	Wheat (*Mistral*)	1290.3	13 May
2	7.59	5157.6	Wheat (*Mistral*)	1316.8	14 May
3	11.77	5534.3	Barley (*Zebra*)	1179.9	12 May
4	11.08	3727.5	Barley (*Zebra*)	1181.3	11 May
5	7.88	4166.9	Barley (*RGT Planet*)	1127.6	16 May
6	13.05	4227.9	Barley (*RGT Planet*)	1117.1	19 May
7	7.61	6668.5	Oats (*Ringsaker*)	1223.4	17 May
8	7.77	5788.2	Barley (*Harbringer*)	1136.1	21 May
9	7.24	6166.0	Oats (*Ringsaker*)	1216.4	18 May

Weather data. The weather data was acquired from the open interface provided by the Finnish Meteorological Institute for Pori area. The thermal growing season started on 13th of April in 2018 and the cumulative temperature was calculated using that as the beginning date. As growth of crops is dictated by the accumulation of sunlight amongst other climatological, soil and nutrient variables, cumulative temperature was deemed robust enough indicator of interval between subsequent data collection days (instead of e.g., time in days). Being a common way to express crop growth phase, the cumulative temperature was utilized as a part of the input data to encode passing of time for the temporal models.

Yield data. As the target data, i.e., the data used as the ground truth for training the models, yield data was acquired during the harvest of each field. The harvesters were equipped with either a Trimble Navigation (Sunnyvale, California, USA) CFX 750 or John Deere (Moline, Illinois, USA) Greenstar 1 yield mapping sensor systems. The systems produce a cloud of geolocated points with multivariate information about the harvest for each point in vector format. This data was first accumulated field-wise and then filtered to contain data points where the yield was between 1500 and 15,000 kg/ha and the speed of the harvester was between 2 and 7 km/h [2]. Finally, the yield map rasters were generated by interpolating the vector points over each field.

2.2. Data Preprocessing

The RGB images taken with the UAV and the cumulative temperatures for imaging dates were utilized as the input data with which the predictions about yields were performed. As spatial models generally have a built in limitation of being able to utilize data with fixed dimensions only, data had to be clipped to smaller fixed dimension frames. As an intended side-effect, using smaller frames makes it possible to better model intra-field yield variability. Like in [2], the fields were split into smaller overlapping frames of size 40 × 40 m with a lateral and vertical step of 10 m. cumulative temperature was added as an additional layer in conjunction with the RGB-layers to have the data contain necessary information for temporal feature learning. The added layer contains constant values corresponding to the field and time of acquisition. The design choice of introducing this data as an additional layer was to have a single source of similarly constructed data for each model architecture.

During the extraction of frames we included every frame that had at least half of its data present at field edges into the final data set. The reasoning behind this was that the spatial models effectively learn filters that are applied over the spatial input data in a successive manner (see Section 2.3.1 for more). Thus, salient features are expected to be present in a frame albiet being just partial due to being located at a field's edge.

The data was also scaled to aid the models in their learning. All values were scaled to the range $[0, 1]$ using feature-wise maximum values as scalers. For the value ranges of unscaled input RGB data, the cumulative temperature calculated from the beginning of the thermal growing season and yield data, see Table 2. As the input data for this study was temporally sequential, the geolocationally matched frames were clipped across every image acquired at a different date for each field. Each sequence of frames was then coupled with geolocationally matching average yield.

Table 2. The value ranges of used input and target variables prior scaling.

Data	Min	Max	Mean	Std
RGB: R	105	254	186.0	19.5
RGB: G	72	243	154.3	18.8
RGB: B	58	223	126.7	18.9
Cumulative °C	388.6	2096	1192	545.0
Yield, kg/ha	1500	14,800	5287	1816

As the last step, the sequences of frames coupled with matching yield information were shuffled and split to training and hold out test data sets with 70%/30% ratio. The samples in the training set are used to optimize the model during the training. The test set is then utilized to evaluate model capabilities with previously unseen data, i.e., its generalization capabilities.

With the total number of generated sequences of frames being 2586, the training data set contained 1810 frame sequences (27,150 frames) and the test set 776 frame sequences (11,640 frames). The general process of generating the frames is depicted in Figure 1.

With the resolution of 0.325 px/m, a single spatial layer in the input data had the dimensions of 128 × 128 px. Using RGB-data with an additional layer constructed from the cumulative temperature conforming to the imaging date, a single frame of data consisted of four layers. With 15 frames, each frame corresponding to a particular week of the growing season, an input sequence of frames thus had the dimensions of $[15 \times 4 \times 128 \times 128]$.

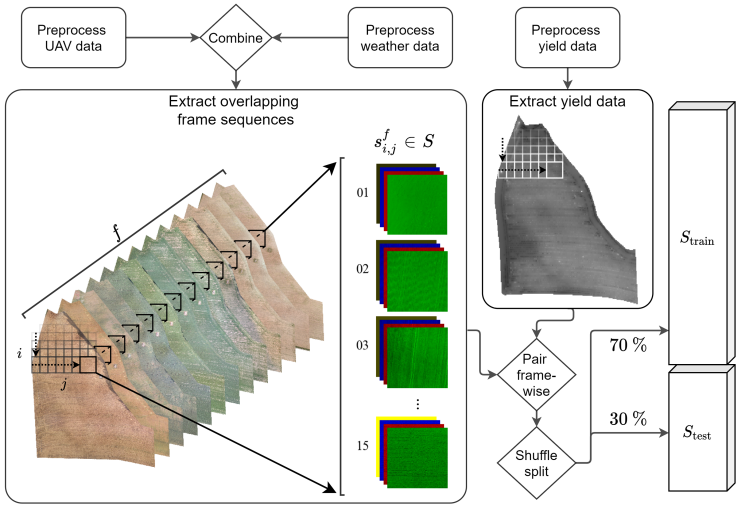

Figure 1. Input frame sequence and target average yield extraction process. Sequences of frames S of fixed width and height were extracted from cumulative temperature enhanced RGB image mosaic sequences as the input data, with f being a distinct field and $s^f_{i,j}$ an extracted sequence of frames from f. The four-layer YBRG, Y being the cumulative temperature, input frames were then geolocationally paired with corresponding yield data to form input-target pairs. Lastly, data was shuffled and split to training and test sets.

2.3. Model Architectures

2.3.1. Convolutional Neural Networks

Convolutional neural networks, often referred to as CNNs, have solidified their place in modeling tasks where the input data is either spatial or spatially representable [16,17]. The main component of the model is the convolution operation, where a set of trainable kernels (or filters) is applied to the input data resulting in a set of spatial features describing the data. For more in-depth explanation of the operations within a single convolution layer, like the application of convolution and pooling, see [2]. The model learns basic features in the first layers and composite features of these basic features at further layers [18]. To help the model better learn these features, batch normalization can be applied to the inputs [19]. The final output of a plain CNN is a set of feature maps. Depending on the use case, these can be either directly utilized or, for example, flattened and fed to a fully connected (FC) layer for regression or classification purposes.

2.3.2. Long Short-Term Memory Networks

The Long Short-Term Memory (LSTM) networks, originally introduced in [20], have been widely utilized in sequence modeling tasks [21]. There are two general concepts to the LSTM that help it in learning temporal features from the data. The first is the concept of memory, introduced as the cell state. The other is the concept of gates, effectively trainable FC layers, manipulating this cell state in response to the new inputs from the data and past outputs of the model. To handle sequences of data, the model loops over the sequences altering its cell (C) and hidden (H) states in the process using the combination of learned parameters in the gates and non-linear activations when combining the gate outputs. Following the Pytorch [22] implementation of LSTM, the following functions are computed:

$$
\begin{aligned}
g_t^i &= \sigma(W_x^I x_t + W_h^I h_{t-1}) \\
g_t^f &= \sigma(W_x^F x_t + W_h^F h_{t-1}) \\
g_t^c &= \tanh(W_x^C x_t + W_h^C h_{t-1}) \\
g_t^o &= \sigma(W_x^O x_t + W_h^O h_{t-1}) \\
C_t &= g_t^f \odot C_{t-1} + g_t^i \odot g_t^c \\
H_t &= O_t = g_t^o \odot \tanh(g_t^c)
\end{aligned}
\tag{1}
$$

where $g_t^{\{i,f,c,o\}}$ are the outputs of the input, forget, cell and output gates, respectively. The gates of the model contain its trainable parameters W. x_t denotes the external input and h_{t-1} the model's previous output. t denotes the current time step. C_t and H_t are the final computed cell and hidden states, respectively. The output O_t of the model is the last computed hidden state H_t. b are the bias factors and $\odot$ is the dot product. The general architecture of an LSTM is depicted in Figure 2.

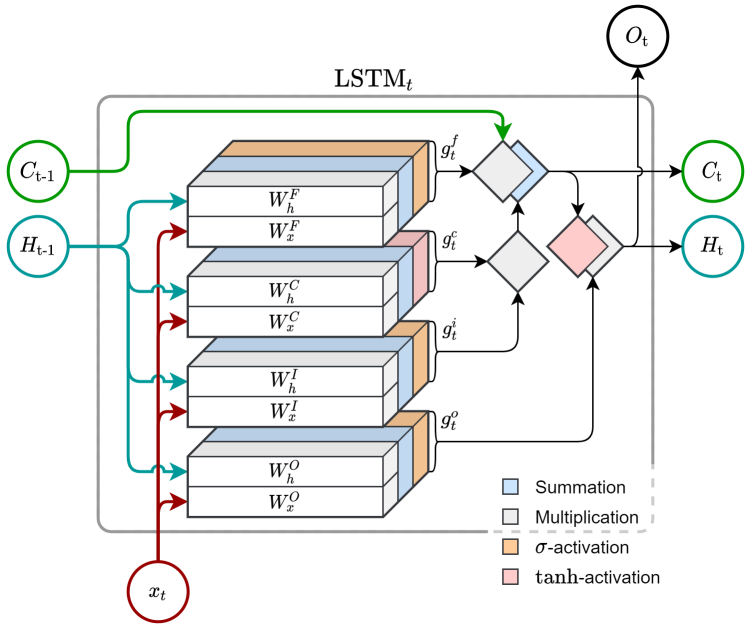

Figure 2. The inner architecture of an LSTM at a time step t. The model takes as its inputs the previous cell state C_{t-1} and hidden state H_{t-1} with the current item x_t of the input sequence. The H_{t-1} and x_t are then passed to forget (W^F), cell (W^C), input (W^I) and output (W^O) gates. These gates, effectively shallow FC layers, are responsible for determining what to keep from previous memory C_{t-1} accumulated from past experiences and what to incorporate to it as the current C_t. This is how the model is able to learn temporal features.

LSTMs can also be employed in bidirectional and stacked form. Bidirectional LSTMs train an additional model in comparison to the unidirectional LSTM presented in Figure 2. One LSTM reads the input from start of the sequence to end ($t_0 \rightarrow t_n$), while the other reads the input from end to start ($t_n \rightarrow t_0$). The outputs of these two parallel models are then combined as final temporal feature outputs [23]. When LSTMs are stacked, the first LSTM operates on the input sequence and subsequent LSTMs then operate on sequences of temporal feature outputs produced by preceding models. Bidirectionality helps

the model learn features from both sides of input sequences, while stacking helps in learning higher level temporal features [24].

2.3.3. CNN-LSTM

The CNN-LSTM is a composite model consisting of a spatial feature extractor or transformer, i.e., a pretrained CNN, and a temporal model, the LSTM [3]. The general idea is to both gain the ability to utilize spatial data and perform sequential modeling with LSTM networks.

The architecture of the pretrained CNN was implemented according to [2] with certain adaptations to have the model better serve as pre-trained spatial feature extractor of the composite CNN-LSTM. Firstly, the model was modified to accept four-band inputs. Secondly, the CNN layers were decoupled from the prediction-producing FC layers as a separate sub-module. Other than those two, the CNN consists of six convolutional layers with batch normalization in every layer and max pooling applied in the first and last layers. All convolution operations utilize 5×5 kernels with 128 kernels in the last operation and 64 in the ones preceding that. The convolutions are performed with zero padding to maintain constant dimensions. The maintaining of dimensions was initially implemented to allow adding an arbitrary number of in-between convolutional layers to the model without diminishing the intermediate hidden output dimensions to oblivion. The input max pooling uses a 8×8 and the output max pooling a 2×2 kernel. The output of the last convolutional layer is passed to a linear layer, squashing the hidden feature space to 256 features akin to [3]. These features are fed to the recurrent LSTM model. When pre-training the CNN only, the output of the squashing linear layer is fed to another linear layer producing the prediction outputs for error metric calculations. This outermost linear layer is omitted when the CNN is used as a part of the CNN-LSTM. The architecture of the spatial feature extracting CNN of the composite CNN-LSTM model is depicted in Figure 3.

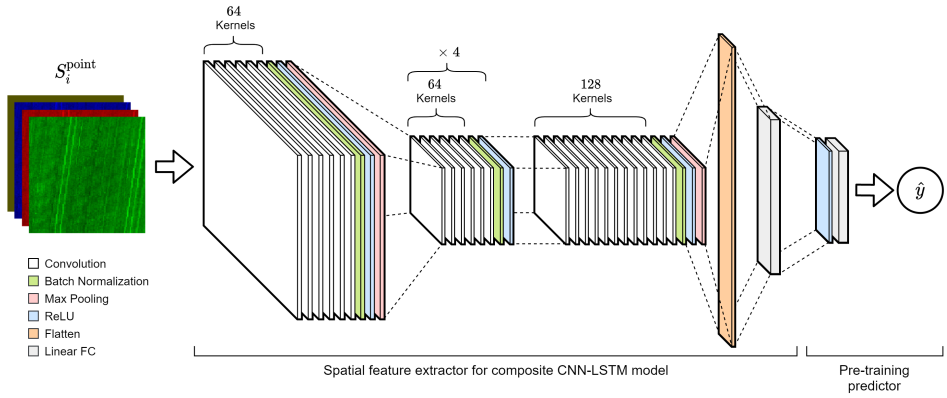

Figure 3. The spatial feature extracting CNN of the CNN-LSTM composite model, i.e., the pretrained CNN. The model is similar to the best performing model of [2]. Alterations in the FC layer composition had to be made to provide sufficient features for the LSTM utilizing the CNN as its input generator.

The temporal feature extracting part of the model is an LSTM, accepting sequences of spatial features as its inputs. During the hyperparameter optimization, we performed architectural experiments also with bidirectional and stacked (multi-layered) LSTMs. Generally, the option to use dropout [25], a regularization technique, is also part of the architectural implementation and that is also the case with Pytorch's LSTM implementation.

While the spatial feature extracting CNN could have been jointly trained with the LSTM, we chose to use a pre-trainend CNN to see whether the point-in-time spatial features could be utilized to perform sequential regression, similar to [10]. We selected this approach also to tie the composite model better

to the framework of the study by [2] by first training the CNN and then examining the ability of the LSTM to learn the temporal features in isolation from the CNN.

2.3.4. ConvLSTM

ConvLSTM [4] is a model combining the features of convolutional and sequential models into a single architecture, using convolutional layers (convolution with pooling etc.) as the LSTM's gate functions. This makes it possible to feed the sequential model the spatial data directly. Akin to how convolutional networks learn, the gates learn to utilize the convolutional kernels to provide the best set of spatial features when building and modifying the cell state C. Thus, contrary to the CNN-LSTM, no pre-extraction of spatial features for further spatial modelling is required. Our implementation of the recurrent architecture follows Equation (1).

From the point of architectural composition, the ConvLSTM is an LSTM at its essential core. In the ConvLSTM, using Figure 2 as a reference, the cell and hidden state altering gates $W^{\{F,C,I,O\}}$ have, however, been changed from conventional LSTM's shallow FC layers to shallow CNNs. To extract robust features from the input data, the gates W_x^* for inputs also employ a max pooling layer with a 3×3 kernel having padding and stride to halve input image dimensions after the first convolution. Due to the nature of CNNs learning spatial features in increasing complexity from layer-to-layer, we also allowed the model to utilize up to two convolutional layers for each W. Like with the CNN of the CNN-LSTM, we wanted to make sure that the intermediate feature map dimensions remaing unchanged, i.e., do not diminish as items in the sequences are processed. Thus, we used 32 convolutional kernels with 5×5 kernel shape and sufficient padding. The possibility to use batch normalization for inputs, stacking, bidirectionality and dropout were also implemented to find the best performing architectural composition.

2.3.5. 3D-CNN

As initially reported by [5], 3D-CNNs performed remarkably well in modeling tasks involving spatio-temporal data. Being CNNs, the 3D-CNNs utilize all same architectural features as more commonly used convolutional models. What's different is their use of convolution in the depth dimension, searching for robust features across sequences of input data in addition to spatial features extracted from the individual images. The sequential nature of input data is not limited to time, but can also be, for example, hyperspectral multi-layer point-in-time data with the aim of finding salient intra-band features [14]. The 3D-convolution is applied with a learnable three dimensional kernel, depicted in Figure 4. Kernel dimensions are in $[Z \times X \times Y]$ format, where Z denotes the time dimension.

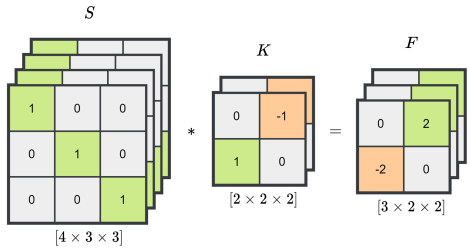

Figure 4. An illustration of 3D convolution. The 3D convolution operation effectively applies the kernel in one additional dimension cmompared to the normal convolution, the depth or z-axis. Like the input, the kernels are three dimensional. The dimensions of the feature map conform to how many times a kernel can wholly be applied to the input data along all three dimensions. With stride of one in each dimension, a $[2 \times 2 \times 2]$ kernel is applied on a $[4 \times 3 \times 3]$ input sequence of layers two times in x and y dimensions and thrice in z dimension, resulting in a $[3 \times 2 \times 2]$ feature map, its values being sums of products over distinct applications of K akin to 2D convolution.

The general architecture of the 3D-CNN we implemented conforms closely to how a CNN is generally constructed with the exception of using 3D instead of 2D convolutions. In the first layer the data is, however, grouped by layers as depicted in Figure 5. This is to have the model learn the spatio-temporal features of the data on a per-layer basis.

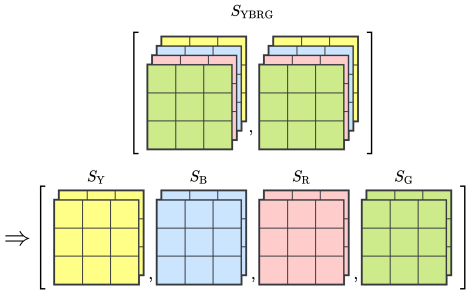

Figure 5. In the first layer of the 3D-CNN, the sequences of multi-layer input data are handled layer-wise. This helps the model first learn layer-wise spatio-temporal features, which are then composed as interlayer spatio-temporal features in the subsequent layers.

Our implementation follows the general CNN architecture of Figure 3. All layers prior to the last have the same number of kernels, the last having twice as many. We employ max pooling only in the first and last layers while the intermediate layers preserve intermediate feature map dimensions. The exact number of kernels is determined via hyperparameter tuning. As per [5], we perform convolutions with $[3 \times 3 \times 3]$ kernels with zero padding, having the pooling layers perform the diminishing of feature map dimensions. Mixing the depth-wise steps from [5] and spatial steps from [2], the first max pooling employs a $[1 \times 8 \times 8]$ kernel while the last a $[2 \times 2 \times 2]$ kernel. The kernels' strides equal respective kernel sizes, i.e., no overlap is applied. Like with the ConvLSTM, we the option to utilize batch normalization in every layer was also implemented for hyperparameter tuning purposes.

2.4. Training and Optimization

The process of training neural networks generally requires hyperparameter tuning. While model parameters, such as the layer-wise weights, are optimized during training in response to regression errors with the selected optimization algorithm, the hyperparameters are what dictate how the model is initialized and in what manner the optimization is applied. Examples of these hyperparameters include the learning rate and model depth. From available hyperparameter tuning methods we chose to use random search, in which a distribution is defined for each hyperparameter and then a value is randomly drawn for each distinct training [26].

We first performed the hyperparameter tuning for the pretrained CNN of the CNN-LSTM. Unlike sequential models, the pretrained CNN was fed single frames (i.e., point-in-time) drawn randomly from the set of all training data set frames. The goal was to have the model learn general spatial features for the whole growing season. Following [2], we used Adadelta [27] as the optimizer. Due to having input data consist of four distinct layers instead of only RGB layers, we performed tuning for the learning rate. Weight decay and the ρ coefficient were utilized from [2].

For the spatio-temporal models, we used Adam [28] as the optimizing algorithm for each model architecture akin to [7,8,11]. The spatio-temporal models were trained with frame sequences. Each model was tuned for LSTM and CNN architectural (where applicable) and optimizer hyperparamenters. The architectural and optimizer hyperparameters are given in Table 3. All hyperparameters were tuned simultaneously and not in sequential succession, meaning that the

hyperparameter values were drawn from their respective distributions for each hyperparameter at the start of a training.

Table 3. The model-specific hyperparameters and their distributions tuned during the random search. Square brackets indicate closed interval with lower and upper limit included, while curly brackets indicate a set from which a value was chosen from. The presence of $\vee$ indicates a boolean toggle with $p = 0.5$. The $\log_{10}$-uniform distribution used for the learning rate draws a value a from a float-uniform distribution in a given range to calculate 10^a.

Hyperparameter	Distribution	Pre-CNN	CNN-LSTM	ConvLSTM	3D-CNN
LSTM Architectural parameters					
LSTM layers	int-uniform	-	$[1, 3]$	$[1, 3]$	-
Dropout	float-uniform	-	$[0, 1] \vee 0$	$[0, 1] \vee 0$	-
Bidirectional	bool		$0 \vee 1$	$0 \vee 1$	-
CNN Architectural parameters					
CNN layers	int-uniform	-	-	$[1, 2]$	$[2, 5]$
Batch normalization	bool	-	-	$0 \vee 1$	$0 \vee 1$
Kernels	set	-	-	$\{32, 64, 128\}$	$\{32, 64, 128\}$
Optimizer parameters					
Learning rate	$\log_{10}$-uniform	$[-4, -1]$	$[-4, -2]$	$[-4, -2]$	$[-4, -2]$
L2-regularization	float-uniform	-	$[0, 1] \vee 0$	$[0, 1] \vee 0$	$[0, 1] \vee 0$

With each sequential model type we performed 300 distinct model training session, using random search for hyperparameter tuning and Skorch [29] as the training framework. For the pretraining of the CNN-LSTM's spatial feature extracting CNN, 50 models were trained to tune the learning rate due to the additional layer in inputs. The ρ-coefficient of the Adadelta algorithm and the weigth decay parameters were utilized from [2]. The total number of trained models was thus 950. The parameters of each model were initialized with *xavier*-uniform intialization [30]. During training, we utilized early stopping with patience for stagnant progress of 50 epochs. A single training iteration was allowed to continue for a maximum of 250 epochs. With continued training, where the best performing model parameter configuration is utilized as the starting point for a subsequent round of training, a model was allowed to be trained a maximum of 500 epochs. However, the use of early stopping was allowed to halt the training prior reaching that limit. Training was conducted with a separate training data set, having the training process utilize 5-fold cross-validation, where the training and validation batches are derived from the training data set. The final evaluation of a trained model was performed with the hold-out test data set. The models were trained in a distributed computation environment, utilizing Nvidia Tesla V100 Volta and Pascal architecture cloud GPUs.

3. Results

From the sets of trained models produced during the hyperparameter tuning process, the best performing models were singled out. During training we monitored the mean squared error (MSE) of the 5-fold cross validation. We also computed metrics for root mean square error (RMSE), mean absolute error (MAE) of unscaled targets, mean absolute percentage error (MAPE) and the coefficient of determination (R^2). The best performing model architecture was the 3D-CNN, expressing notably better performance with the best performing model than the rest of the trained architectures. The model performing worst was somewhat surprisingly the ConvLSTM, showing performance inferior even to the pretrained CNN trained with just point-in-time data. The performance metrics for the unscaled predicted and true target values for each model architecture are given as in Table 4.

Table 4. The performance metrics of the best-performing models resulting from model-specific hyperparameter tuning process with samples from the test set. The trained models were evaluated with a hold-out test set. Best performance was achieved with the 3D-CNN architecture. The number of trainable parameters indicate the model complexity. Best performance values are in bold text.

Model	Test RMSE (kg/ha)	Test MAE (kg/ha)	Test MAPE (%)	Test R^2 -	Trainable Parameters
Pretrained CNN	692.8	472.7	10.95	0.780	2.72×10^6
CNN-LSTM	456.1	329.5	7.97	0.905	2.94×10^6
ConvLSTM	1190.3	926.9	22.47	0.349	9.03×10^5
3D-CNN	**289.5**	**219.9**	**5.51**	**0.962**	7.48×10^6

The most consistently fitting sequential architecture was the CNN-LSTM in terms of test set performance with trained models. Other architectures produced occasional ill-fitted models with errors several magnitudes higher than their best performing counterparts. The RMSE percentiles depicting the general consistency in fitting for the spatio-temporal models are given in Table 5.

Table 5. The RMSE percentiles accross all trained spatio-temporal models. The RMSE percentiles indicate the consistency of a model architecture in generalizing to unseen samples with the training data. Out of the three, the CNN-LSTM was most consistent in how it was able to fit to the data and produce generalizable results. The training of other model architectures produced occasionally ill-fitted models. Best performance values are in bold text.

Model	Test RMSE (kg/ha)				
	Min	25%	50%	75%	Max
CNN-LSTM	456.1	**655.1**	**1475.6**	**1623.7**	$\mathbf{2.152 \times 10^3}$
ConvLSTM	1190.3	1477.8	1646.6	8750.2	1.334×10^6
3D-CNN	**289.5**	1355.4	1493.6	1649.0	1.926×10^6

Due to the training of the model architectures being a process of empirically evaluating randomly drawn hyperparameter sets, visualization of the hyperparameters against a performance metric further helps in understanding model fitting consistency. Out of the architectures, the CNN-LSTM and the 3D-CNN show similar behaviour in hyperparameter value distribution, the latter having a discernible dispersion in the values against the performance metric. ConvLSTM, as already stated, exhibits clearer sporadicity. The architecture-specific hyperparameter distributions plotted against the test RMSE are given in Figure 6.

The best performing configuration of hyperparameters dictating how a model is to be initialized and trained were sought by performing random search. In random search, each hyperparameter is assigned with a distribution, from which a value is drawn for each independent training of the model. The hyperparameters for the best performing models are given in Table 6.

In addition to performing comparative performance evaluation between the selected deep learning architectures with data sequences spanning the time from sowing to harvest, we also evaluated the performance of the best performing model configuration (architecture with hyperparameters) using data from an actionable time frame. In other words, we combined various configurations of input data sequences starting from image data acquisition dates closest to sowing (week 21) and ending at the midsummer (week 25). The following sequence configurations were built using the aforementioned time range:

- Weeks 21, 22, 23, 24, 25; five temporal frames.
- Weeks 21, 22, 23, 24; four temporal frames.
- Weeks 22, 23, 24, 25; four temporal frames.

- Weeks 21, 22, 23; three temporal frames.
- Weeks 23, 24, 25; three temporal frames.
- Weeks 21, 23, 25; three temporal frames.

We trained ten iterations of the best model configuration, the 3D-CNN, for each input sequence type to account for the effects of random model parameter initialization. The training was conducted as before, utilizing 5-fold cross-validation with the training data and testing the generalization capabilities with the hold-out test data, separately. The performance of these trained models with the test data are given in Table 7, where each row corresponds to a distinct configuration of input frame sequences. The best perfoming configuration in terms of RMSE and MAE is the four week long sequence taken from the beginning of the season (weeks 21 to 24). In terms of MAPE, the best performing configuration, however, consists of five weeks from the beginning of the season (weeks 21 to 25), although the difference to the four week sequence is small.

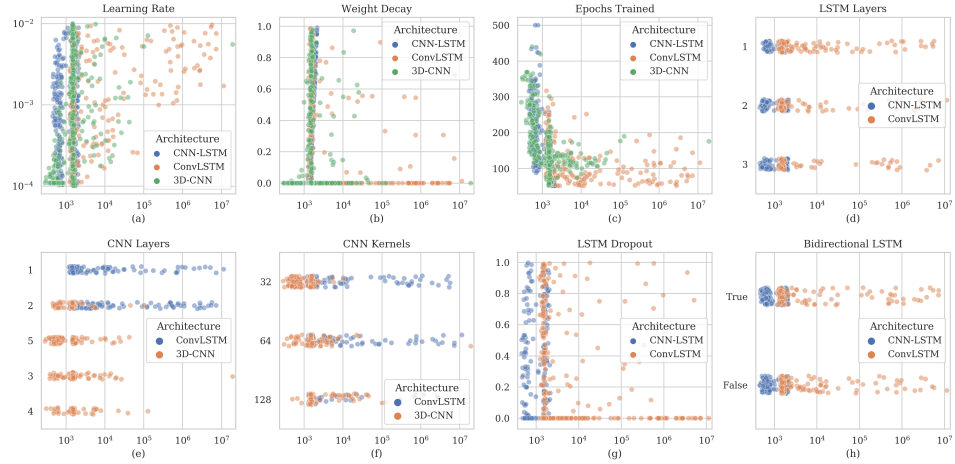

Figure 6. The architecture-specific distributions of hyperparameters against the test RMSE (x axis). (**a**,**b**) are the optimizer hypeparameters and (**c**) is the training length in epochs. (**d**,**g**,**h**) are the LSTM architectural hyperparameters, while (**e**,**f**) are the CNN architectural hyperparameters. (**d**,**e**,**f**,**h**) contain categorical values in y axis with values spread category-wise for easier observation of clustering. The rest of the sub-figures have a continuous y axis.

Table 6. The architecture specific hyperparameter values for the best performing models. The value types conform to the values given in Table 3. The feature extracting CNN of the CNN-LSTM was not tuned for hyperparameters as tuning results from previous study were utilized.

Hyperparameter	**Pre-CNN**	**CNN-LSTM**	**ConvLSTM**	**3D-CNN**
	LSTM Architectural parameters			
LSTM layers	-	2	2	-
Dropout	-	0.5027	0.9025	-
Bidirectional	-	0	1	-
	CNN Architectural parameters			
CNN layers	6 *	-	1	5
Batch normalization	Yes *	-	No	No
Kernels	128/64 *	-	32	32
	Optimizer parameters			
Learning rate	1.000×10^{-1}	7.224×10^{-4}	1.361×10^{-3}	1.094×10^{-4}
L2-regularization	0.9 *	0.0	0.0	0.0

* Values taken from [2].

Table 7. Retraining results of the best performing 3D-CNN configuration with various input sequence configurations from the test set. The input data was constructed from the first five imagings (weeks 21 to 25). The composition of weekly data was varied and variations evaluated by fitting the best performing 3D-CNN configuration to each variation. Best performance values are in bold text.

Weeks in Input Sequence	Test RMSE (kg/ha)	Test MAE (kg/ha)	Test MAPE (%)	Test R^2 -
21, 22, 23, 24, 25	413.8	320.6	**7.04**	0.921
21, 22, 23, 24	**393.9**	**292.8**	7.17	**0.929**
22, 23, 24, 25	439.3	343.0	7.90	0.911
21, 22, 23	543.5	421.4	10.02	0.864
23, 24, 25	425.0	326.6	8.25	0.917
21, 23, 25	478.1	369.3	8.72	0.895

Operating with single frames, the models can be used to construct predictions for whole fields. This is achieved by extracting frames from an image of the fields and feeding them as inputs to the model. Re-arranging the predictions to original field shape yields a map of frame-wise yield predictions. The performance of the best performing 3D-CNN configuration with both full length and shortened sequences is illustrated in Figure 7 with a 10 m step between predicted points. As the test set was constructed from frame sequences randomly taken from all extracted frame sequences, the illustrations contain frames from both the training and the test set.

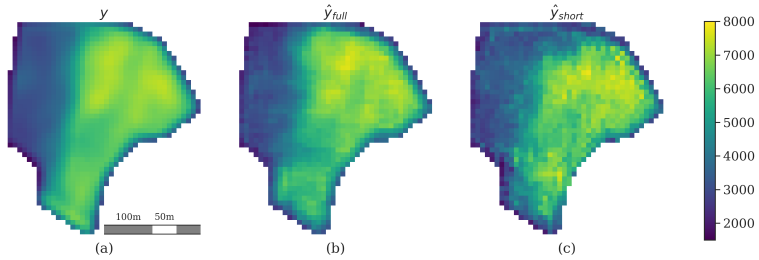

Figure 7. Frame-based 3D-CNN model performances against true yield data. (**a**) is the true yield map of the field. (**b**) is the modelled prediction, utilizing the full length frame sequences. (**c**) is then the actual in-season prediction utilizing four first frames of the weekly frame sequence. Units are absolute values in crop yield kg/ha. One pixel in the images corresponds to a 10 × 10 m area. Images are unsmoothed, represent the values as they were produced and contain samples from both training and test sets due to how the sets were constructed.

4. Discussion

In this study we evaluated the feasibility of using spatio-temporal deep learning architectures in modelling crop yield at the intra-field scale. Using sequences of UAV and weather data collected in the vicinity of Pori, Finland, during the growing season of 2018, we split the fields to geolocationally matched temporal sequences of frames of fixed width and height. We developed and trained three different model architectures: CNN-LSTM, ConvLSTM and 3D-CNN. We first determined the best performing architecture by performing hyperparameter tuning with complete temporal sequences of frames (15 time steps). With the best performing model architecture and hyperparameter configuration we then evaluated the predictive capabilities of the models by using a shorter temporal sequence of frames from the beginning of the growing season.

Of the architectures, the 3D-CNN performed the best in full sequence modelling. The best performing model consisted of five 3D-CNN layers using 32 kernels in the layers. Other architectural configurations are given in Section 2.3.5. The model attained 218.9 kg/ha test MAE, 5.51% test MAPE and 0.962 test R^2-score. Compared to the study presented in [2] using just a point-in-time single

frame predictor with 484.3 kg/ha MAE and 8.8% MAPE, the modelling performance was improved by 265.4 kg/ha MAE (54.8% improvement) and 3.29% MAPE (37.4% improvement). In terms of prediction performance with smaller sequences from the beginning of the season, the 3D-CNN performed the best using four first frames of the whole input sequences. With a shorter sequence the model attained 292.8 kg/ha test MAE, 7.17% test MAPE and 0.929 test R^2-score. The respective improvements to the best performing model presented in [2] were 191.5 MAE (39.5% improvement) and 1.63% MAPE (18.5% improvement).

Recent studies make use of UAVs in a variety of imaging applications. The use of UAVs has become more common, as shown in [31], a review of UAV thermal imagery applications in the domain of precision agriculture. One of the reasons is the increased need of performing classification and regression at scales more accurate than what is attainable by publicly available satellite data sources. However, the use of point-in-time UAV data is common. Ref. [32] utilized UAVs to gather hyperspectral data of potato tuber growth at the resolution of 2.5 cm/px. They utilized traditional ML methods, such as linear models and decision trees, to perform tuber yield estimation using individual data points gathered in-season at the intra-field scale, achieving 0.63 R^2-score for the tuber yield prediction accuracy with a Ridge regression. Ref. [33] used UAV to collect multispectral data from wheat and corn fields to estimate intra-field crop nitrogen content using linear regression and point samples—spatial features were not utilized. They fit multiple linear models to wheat and corn and attained 0.872 R^2-score on average. Ref. [34] performed wheat leaf area index and grain yield estimation with various vegetation indices derived from point-in-time multispectral UAV data using multiple machine learning methods, neural networks included. The highest performance they attained was 0.78 R^2-score with a Random Forest. However, they fed the input data as point samples.

Satellites perform frequent overflights over vast areas accross the globe. They are thus an ideal source of automatically generated multi-temporal remote sensing data [35]. This is one of the reasons, why spatio-temporal modelling is more notably present in the context of publicly available satellite data sources, contrary to UAV data requiring manual collection. Spatio-temporal models akin to the setting of our study have been utilized in various modelling tasks with remote sensing data in the domain of agriculture. Performing county-scale soybean yield prediction, ref. [6] used a CNN, an LSTM and a composite CNN-LSTM to model soybean yield with in-season satellite data. They achieved an average 0.78 R^2-score with the spatio-temporal CNN-LSTM model. Their input data resolutions were from 500 × 500 m/px to 1 × 1 km/px. Ref. [7] performed crop type classification in Europe and Africa with multi-temporal satellit data at resolutions from 3 × 3 m/px to 10 × 10 m/px. They attained F1 scores 91.4 for the CNN-ConvLSTM and 90.0 for the 3D-CNN, averaged over crop types in their Germany data set. Ref. [8] performed pre-season crop type mapping for the area of Nebraska, US, employing a CNN-ConvLSTM to extract spatio-temporal features from multi-temporal multi-satellite composite data set. Using prior years of crop type related data to predict a map of crop types, they attained an average accuracy of 77% across all crop types in their data. The data was processed to a resolution of 30 × 30 m/px. Ref. [9] utilized a 3D-CNN to classify crop types from multi-temporal satellite data gathered from an area within China, acquiring a classification accuracy of 98.9% with the model. Their input data resolutions were from 4 × 4 m/px to 15 × 15 m/px. Ref. [36] performed weekly UAV image collections in a controlled field experiment with soybeans, performing seed yield prediction with multiple linear models fit the multi-temporal data. Thus, spatio-temporal modelling with novel techniques was not performed. With seed yield prediction, they achieved 0.501 adjusted R^2 score. The resolution of their data was 1.25 × 1.25 cm/px.

In remote sensing, the multitemporal aspect of satellite sensor data the has been well studied. In their review of the applications of multisource and multitemporal data fusion in remote sensing, ref. [37] show how studies utilizing the temporal feature of satellite data are rather common. However, in terms of models and data usage settings, they only briefly mention how novel deep learning architectures have only recently been applied in this data domain. They cite that both data and methods, especially the latter, are still under development and a subject of further research. While

some studies have not found additional benefit in using multitemporal data [38], partly due to selected data utilization techiniques, others find benefit over using just point-in-time data [33,35–37].

Regarding the poor performance of the ConvLSTM in our study, the studies by [7,8] might provide some basis for understanding the phenomenon. In both studies, the ConvLSTM was preceded with an exclusively spatial feature extracting CNN model. The extracted feature maps were then fed to the ConvLSTM for temporal feature extraction. While in our study we experimented with multiple convolutional layers in the ConvLSTM model, it could very well be that using a pre-trained CNN akin to CNN-LSTM is required for the ConvLSTM as well. Model complexity is another way to look at this, as the 3D-CNN model was more complex compared to the ConvLSTM. This indicates that the effective capacity of the ConvLSTM might indeed be too low. Thus, increasing the effective capacity by either adding a spatial feature pre-extractor or increasing the gate-wise layer count could increase the performance of this model architecture in similar study setting.

As we utilized weather information at city-scale, the precision of change in the growth phase could be further improved with specifically located weather stations. Weather stations located in the approximate vicinity of the fields under scrutiny could provide better and more accurate measurements of the local temperatures and other climatological variables and thus might help the model produce even better predictions when sequences are involved. Using other data sources, such as soil information and topology maps, could also be further utilized to improve the predictive capabilities of the model. As growing season provides information about how the crops have concretely developed, the soil and topology maps provide more in terms of a prior that the UAV images are then used to further develop as new samples emerge.

A limitation to our study is the use of aggregated crop type data collected from various fields. Using a single model to predict for wheat, barley and oats prohibits both the inference with and the performance analysis of the model on a per-crop basis. Additionally, the remote sensing data based modelling approach doesn't take into account any existing crop growth models. Those could well be utilized to further provide better performance, akin to what has been done in [36], but this is outside the scope of our study. That being said, the modelling task of this study was not that of crop growth, but yield estimation with UAV remote sensing data.

5. Conclusions

Our study seeks to combine three increasingly common but yet seldom co-utilized concepts in the domain of crop yield estimation: the use of high resolution UAV image data, time series regression and novel spatio-temporal neural network architectures. It has already been shown that crop yield prediction with spatial neural networks, i.e., CNNs, is feasible and produces results accurate enough for performing actions in-season [2]. In this study, we show that adding time as an additional feature not only improves the modelling performance with UAV RGB data (see Table 4) but also improves the predictive capabilities (see Table 7). Furthermore, using weekly UAV data gathered during the first month provides enough data for the model to build an accurately predicted yield map from which to draw further conclusions.

To conclude, the use of multitemporal remote sensing data is not only common but also beneficial in crop yield modelling and prediction. Furthermore, the easy accessibility of commercially available UAVs with mounted RGB sensors enables image data acquisition in higher resolutions compared to satellites. This in turn opens up the possibilities to perform modelling and predictions at intra-field scale. As shown in our study, the use of UAV-based data and proper spatio-temporal deep learning techniques is an enabler of more sophisticated Decision Support Systems in the domain of agriculture.

Author Contributions: Conceptualization, P.N. and N.N.; methodology, P.N. and N.N.; software, P.N.; validation, P.N., N.N. and T.L.; formal analysis, P.N. and N.N.; investigation, P.N.; resources, P.N., N.N. and P.L.; data curation, P.N., N.N. and P.L.; writing—original draft preparation, P.N.; writing—review and editing, P.N., N.N. and T.L.; visualization, P.N.; supervision, N.N. and T.L.; project administration, P.N., N.N. and T.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by Mtech Digital Solution Oy, Vantaa, Finland.

Acknowledgments: We would like to thank Tampere University for providing the computational resources and MIKÄ DATA project for providing us with the data.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Narra, N.; Nevavuori, P.; Linna, P.; Lipping, T. A Data Driven Approach to Decision Support in Farming. In *Information Modelling and Knowledge Bases XXXI*; IOS Press: Amsterdam, The Netherlands, 2020; Volume 321, pp. 175–185, doi:10.3233/FAIA200014.
2. Nevavuori, P.; Narra, N.; Lipping, T. Crop yield prediction with deep convolutional neural networks. *Comput. Electron. Agric.* **2019**, *163*, 104859, doi:10.1016/j.compag.2019.104859.
3. Sainath, T.N.; Vinyals, O.; Senior, A.; Sak, H. Convolutional, Long Short-Term Memory, fully connected Deep Neural Networks. In Proceedings of the 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brisbane, Australia, 19–24 April 2015; pp. 4580–4584.
4. Shi, X.; Chen, Z.; Wang, H.; Yeung, D.Y.; Wong, W.K.; Woo, W.C. Convolutional LSTM Network: A Machine Learning Approach for Precipitation Nowcasting. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 802–810.
5. Tran, D.; Bourdev, L.; Fergus, R.; Torresani, L.; Paluri, M. Learning spatiotemporal features with 3D convolutional networks. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 4489–4497.
6. Sun, J.; Di, L.; Sun, Z.; Shen, Y.; Lai, Z. County-Level Soybean Yield Prediction Using Deep CNN-LSTM Model. *Sensors* **2019**, *19*, 4363, doi:10.3390/s19204363.
7. Rustowicz, R.; Cheong, R.; Wang, L.; Ermon, S.; Burke, M.; Lobell, D. Semantic Segmentation of Crop Type in Africa: A Novel Dataset and Analysis of Deep Learning Methods. In Proceedings of the CVPR Workshops, Long Beach, CA, USA, 16–20 June 2019; pp. 75–82.
8. Yaramasu, R.; Bandaru, V.; Pnvr, K. Pre-season crop type mapping using deep neural networks. *Comput. Electron. Agric.* **2020**, *176*, 105664, doi:10.1016/j.compag.2020.105664.
9. Ji, S.; Zhang, C.; Xu, A.; Shi, Y.; Duan, Y. 3D Convolutional Neural Networks for Crop Classification with Multi-Temporal Remote Sensing Images. *Remote Sens.* **2018**, *10*, 75, doi:10.3390/rs10010075.
10. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. In Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015), San Diego, CA, USA, 7–9 May 2015.
11. Liu, Q.; Zhou, F.; Hang, R.; Yuan, X. Bidirectional-Convolutional LSTM Based Spectral-Spatial Feature Learning for Hyperspectral Image Classification. *Remote Sens.* **2017**, *9*, 1330, doi:10.3390/rs9121330.
12. Ienco, D.; Interdonato, R.; Gaetano, R.; Ho Tong Minh, D. Combining Sentinel-1 and Sentinel-2 Satellite Image Time Series for land cover mapping via a multi-source deep learning architecture. *ISPRS J. Photogramm. Remote Sens.* **2019**, *158*, 11–22, doi:10.1016/j.isprsjprs.2019.09.016.
13. Yue, Y.; Li, J.H.; Fan, L.F.; Zhang, L.L.; Zhao, P.F.; Zhou, Q.; Wang, N.; Wang, Z.Y.; Huang, L.; Dong, X.H. Prediction of maize growth stages based on deep learning. *Comput. Electron. Agric.* **2020**, *172*, 105351, doi:10.1016/j.compag.2020.105351.
14. Li, Y.; Zhang, H.; Shen, Q. Spectral–Spatial Classification of Hyperspectral Imagery with 3D Convolutional Neural Network. *Remote Sens.* **2017**, *9*, 67, doi:10.3390/rs9010067.
15. Barbosa, A.; Trevisan, R.; Hovakimyan, N.; Martin, N.F. Modeling yield response to crop management using convolutional neural networks. *Comput. Electron. Agric.* **2020**, *170*, doi:10.1016/j.compag.2019.105197.
16. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90, doi:10.1145/3065386.
17. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9, doi:10.1109/CVPR.2015.7298594.
18. Zeiler, M.D.; Fergus, R. Visualizing and Understanding Convolutional Networks. In *Computer Vision—ECCV 2014*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2014; Volume 8689, pp. 818–833.
19. Ioffe, S.; Szegedy, C. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. *arXiv* **2015**, arXiv:1502.03167.

20. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780, doi:10.1162/neco.1997.9.8.1735.
21. Schmidhuber, J. Deep Learning in Neural Networks: An Overview. *Neural Netw.* **2014**, *61*, 85–117, doi:10.1016/j.neunet.2014.09.003.
22. Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic differentiation in PyTorch. In Proceedings of the NIPS-W, Long Beach, CA, USA, 4–9 December 2017.
23. Schuster, M.; Paliwal, K.K. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.* **1997**, *45*, 2673–2681, doi:10.1109/78.650093.
24. Graves, A. Generating Sequences with Recurrent Neural Networks. *arXiv* **2013**, arXiv:1308.0850, doi:10.1145/2661829.2661935.
25. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958, doi:10.1214/12-AOS1000.
26. Bergstra, J.; Bengio, Y. Random Search for Hyper-Parameter Optimization. *J. Mach. Learn. Res.* **2012**, *13*, 281–305, doi:10.1162/153244303322533223.
27. Zeiler, M.D. ADADELTA: An Adaptive Learning Rate Method. *arXiv* **2012**, arXiv:1212.5701, doi:10.1145/1830483.1830503.
28. Kingma, D.P.; Ba, J. Adam: A Method for Stochastic Optimization. *ICLR* **2014**, 1–15, doi:10.1145/1830483.1830503.
29. Tietz, M.; Fan, T.J.; Nouri, D.; Bossan, B.; Skorch Developers. Skorch: A Scikit-Learn Compatible Neural Network Library That Wraps PyTorch. 2017. Available online: https://skorch.readthedocs.io/ (accessed on 16 October 2020).
30. Glorot, X.; Bengio, Y. Understanding the difficulty of training deep feedforward neural networks. *J. Mach. Learn. Res.* **2010**, *9*, 249–256.
31. Messina, G.; Modica, G. Applications of UAV thermal imagery in precision agriculture: State of the art and future research outlook. *Remote Sens.* **2020**, *12*, 1491, doi:10.3390/RS12091491.
32. Sun, C.; Feng, L.; Zhang, Z.; Ma, Y.; Crosby, T.; Naber, M.; Wang, Y. Prediction of end-of-season tuber yield and tuber set in potatoes using in-season uav-based hyperspectral imagery and machine learning. *Sensors* **2020**, *20*, 5293, doi:10.3390/s20185293.
33. Lee, H.; Wang, J.; Leblon, B. Intra-Field Canopy Nitrogen Retrieval from Unmanned Aerial Vehicle Imagery for Wheat and Corn Fields. *Can. J. Remote Sens.* **2020**, *46*, 454-472, doi:10.1080/07038992.2020.1788384.
34. Fu, Z.; Jiang, J.; Gao, Y.; Krienke, B.; Wang, M.; Zhong, K.; Cao, Q.; Tian, Y.; Zhu, Y.; Cao, W.; et al. Wheat growth monitoring and yield estimation based on multi-rotor unmanned aerial vehicle. *Remote Sens.* **2020**, *12*, 508, doi:10.3390/rs12030508.
35. Liu, S.; Marinelli, D.; Bruzzone, L.; Bovolo, F. A review of change detection in multitemporal hyperspectral images: Current techniques, applications, and challenges. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 140–158, doi:10.1109/MGRS.2019.2898520.
36. Borra-Serrano, I.; Swaef, T.D.; Quataert, P.; Aper, J.; Saleem, A.; Saeys, W.; Somers, B.; Roldán-Ruiz, I.; Lootens, P. Closing the phenotyping gap: High resolution UAV time series for soybean growth analysis provides objective data from field trials. *Remote Sens.* **2020**, *12*, 1644, doi:10.3390/rs12101644.
37. Ghamisi, P.; Rasti, B.; Yokoya, N.; Wang, Q.; Hofle, B.; Bruzzone, L.; Bovolo, F.; Chi, M.; Anders, K.; Gloaguen, R.; et al. Multisource and multitemporal data fusion in remote sensing: A comprehensive review of the state of the art. *IEEE Geosci. Remote Sens. Mag.* **2019**, *7*, 6–39, doi:10.1109/MGRS.2018.2890023.
38. Hauglin, M.; Ørka, H.O. Discriminating between native norway spruce and invasive sitka spruce—A comparison of multitemporal Landsat 8 imagery, aerial images and airborne laser scanner data. *Remote Sens.* **2016**, *8*, 363, doi:10.3390/rs8050363.

PUBLICATION

V

Assessment of Crop Yield Prediction Capabilities of CNN usign Multisource Data

P. Nevavuori, N. Narra, P. Linna and T. Lipping

*Title missing*2021. Accepted for publication

Noname manuscript No.
(will be inserted by the editor)

Assessment of Crop Yield Prediction Capabilities of CNN using Multisource Data

Petteri Nevavuori · Nathaniel Narra · Petri Linna · Tarmo Lipping

Received: date / Accepted: date

Abstract The growing abundance of digitally available spatial, geological and climatological data opens up new opportunities for agricultural data based input-output modeling. In our study, we took a Convolutional Neural Network model previously developed on Unmanned Aerial Vehicle (UAV) image data only and set out to see whether additional inputs from multiple sources would improve the performance of the model. Using the model developed in a preceding study, we fed field-specific data from the following sources: near-infrared data from UAV overflights, Sentinel-2 multispectral data, weather data from locally installed Vantage Pro weather stations, topographical maps from National Land Survey of Finland, soil samplings and soil conductivity data gathered with a Veris MSP3 soil conducitivity probe. Either directly added or encoded as additional layers to the input data, we concluded that additional data helps the spatial point-in-time model learn better features, producing better fit models in the task of yield prediction. With data of four fields, the most significant performance improvements came from using all input data sources. We point out, however, that combining data of various spatial or temporal resolution (i.e., weather data, soil data and weekly acquired images, for example) might cause data leakage between the training and testing data sets when training the CNNs and, therefore, the improvement rate of adding additional data layers should be interpreted with caution.

Keywords crop yield prediction · CNN · multisource input · remote sensing · intra-field

P. Nevavuori
Mtech Digital Solutions Oy,
E-mail: petteri.nevavuori@mtech.fi

P. Linna · N. Narra · T. Lipping
Tampere University,
E-mail: {nathaniel.narra, petri.linna, tarmo.lipping}@tuni.fi

1 Introduction

The application of novel and performant deep learning techniques has seen an increasing trend in the last few years in the domain of Smart Farming and Precision Agriculture [6]. Multiple factors are at play: the abundance of open access satellite system spatial data, availability of commercial unmanned aerial vehicles (UAVs) mountable with external sensors, developments in the soil sensor and camera sensor technologies and the constant need to optimize the production of farms.

Convolutional Neural Networks (CNN), being a subset of deep learning, have been utilized in recent studies on crop yield prediction [6]. The spatial model architecture has been used in predicting cotton yield from RGB data taken at close proximity [15], cereal crop yield prediction from mid-altitude UAV RGB data [11], rice grain yield estimation [18] and crop yield prediction using multisource inputs on patch-scale [4]. In [11] we compared intra-field crop yield estimation performance with NDVI and RGB data from the earlier and later part of the growing season with a variety of CNN configurations. The focus of that study was to assess the generalization capability of a yield prediction model with UAV RGB data.

1.1 Objectives

In this study, we examine the effect of additional field-related spatial or spatial-like data on the intra-field crop yield prediction capabilities using data gathered from the earlier half of the growing season of 2018 (weeks 21 to 26). The objective of this study is to assess crop yield prediction capabilities with the best CNN model composition from [11] by varying the input data configuration. The focus of this study is to see whether additional data, such as weather data, soil and ground information and open-access Sentinel-S2 data would improve the point-in-time prediction performance compared to just using UAV-based RGB data. To limit the scope of the study, architectural and hyperparameter tuning of the CNN model is not addressed here to better isolate performance changes to data and the tuned out architectural and optimizer related hyperparameters were thus taken from [11].

2 Material and Methods

2.1 Data Acquisition

For this study, four crop fields were selected for data acquisition in the vicinity of Pori, Finland (61°29'6.5"N, 21°47'50.7"E) for the growing season of 2018. The field information is provided in Table 1. Following the conclusions of [11], only data from the earlier half of the growing season was considered for UAV and Sentinel-S2 data.

Table 1 The fields selected for the study in the proximity of Pori, Finland. The thermal time is calculated as the cumulative sum of temperature between the sowing and harvest dates. Mean yield has been calculated from processed yield sensor data for each field.

Field #	Size (ha)	Mean yield (kg/ha)	Crop (*Variety*)	Thermal time (°C)	Sowing date
1	7.59	5157.6	Wheat (*Mistral*)	1316.8	14 May
2	11.77	5534.3	Barley (*Zebra*)	1179.9	12 May
3	7.88	4166.9	Barley (*RGT Planet*)	1127.6	16 May
4	7.24	6166.0	Oats (*Ringsaker*)	1216.4	18 May

Table 2 General information of data sources and their original formats.

Source	Type	Resolution/Step	Multitemporal
UAV	Raster	0.3125 m/px	Yes
Sentinel-S2	Raster	[10,20,60] m/px	Yes
Soil samples	Vector	50 m	No
Veris MSP3	Vector	20 m	No
Topography	Vector	2 m	No
Weather	Tabular	-	Yes
Yield	Vector	Varying	No

The multisource input data for the fields consists of UAV-based RGB images, location data, multispectral Sentinel-2 [1] satellite data, sparsely collected and analyzed soil samplings, machine-collected soil information, topography information and local weather station data. General information about the original data sources are given in Table 2. Some of the data were collected during the growing season of 2018 either manually or automatically, while other data were acquired within one year time difference from the aforementioned season. A total of 39 layers constitute the input data sets, while a single layer, the crop yield, is used as the ground truth. These data are described next and the data layers are numbered for further reference.

2.1.1 UAV

It has already been demonstrated that UAV-based RGB data from the first half of the growing season works better than the data from the second half of the growing season and better than NIR only in crop yield prediction [11]. The UAV data of this study has also been used in [10]. The images were taken at average height of 150 meters with a minimum of three ground control points for geometric calibration. Color correction was performed pre-flight and illumination sensors were used for radiometric calibration. We selected UAV-based RGB data acquired for the first weeks after sowing (weeks 21 to 26 of 2018). Thus, every imaged field has five distinct UAV RGB rasters in the collected data set. The data were acquired with overfligths using a SEQUIOA (Parrot Drone SAS, Paris, France) multispectral camera mounted on a Airinov

Solo 3DR (Parrot Drone SAS, Paris, France) UAV. Field-wise orthomosaics were constructed with Pix4D (Pix4D S.A., Prilly, Switzerland) software. UAV data contains the following layers:

1. Red
2. Green
3. Blue

2.1.2 Sentinel-S2

The Sentinel-S2 satellite data for the fields was acquired from the Copernicus Open Access Hub (European Space Agency, Paris, France). The data were date-matched to UAV images during acquisition, prioritizing images where the algorithmically determined cloud probability was lowest. Thus, five Sentinel-S2 rasters with temporal spacing similar to the UAV data were selected for the data set. With the abbreviated names of product layers in brackets, the Level-2A Sentinel-S2 consists of the following layers:

4. Wavelength 0.443 μm (B01)
5. Wavelength 0.490 μm (B02)
6. Wavelength 0.560 μm (B03)
7. Wavelength 0.665 μm (B04)
8. Wavelength 0.705 μm (B05)
9. Wavelength 0.740 μm (B06)
10. Wavelength 0.783 μm (B07)
11. Wavelength 0.842 μm (B08)
12. Wavelength 0.865 μm (B8A)
13. Wavelength 0.945 μm (B09)
14. Wavelength 1.610 μm (B11)
15. Wavelength 2.190 μm (B12)
16. Aerosol optical thickness at 550 nm (AOT)
17. Scene classification layer (SCL)
18. Water vapour map (WVP)
19. Cloud probability (CLDPRB)
20. True color, red (TCIR)
21. True color, green (TCIG)
22. True color, blue (TCIB)

2.1.3 Soil samples

Soil samples were manually collected from the fields by ProAgria, an agronomic counseling instution, and sent to a Eurofins (Eurofins Viljavuuspalvelu, Mikkeli, Finland) laboratory for further analysis. Soil samples were collected with 50 m steps so that a single sample represented an area of 50 $\times$ 50 m. The samples were collected manually once during November 2018. Being point vectors, the data were rasterized with the `gdal_warp` program of the GDAL utility [17]. Soil sample data contains the following layers:

23. Calcium
24. Copper
25. Potassium
26. Magensium
27. Manganese
28. Phosphorus
29. Sulfur
30. Zink

2.1.4 Veris MSP3

To get a finer map of soil chracteristics, a MSP3 soil scanner (Veris Technologies, Salina, Kansas, USA) was used to map the fields at depths of 0-30 cm and 30-90 cm. The measurements were performed during April and May of 2019. The MSP3 measures the soil's electrical conductivity (EC), which is an indicator of soil compactness, wetness and soil type proportions. Additionally, the instrument measures the pH of the soil. Being irregularly spaced point data initially, data had to be rasterized from point vectors. The rasterization was done with the `gdal_warp` program of the GDAL utility [17]. Each field was measured once. Veris MSP3 data contains the following layers:

31. Shallow EC
32. Deeper EC
33. Ratio, (EC SH / EC DP)
34. Infra-red reflectance
35. Red reflectance
36. Soil pH

2.1.5 Topography

The National Land Survey of Finland conducts light detection and ranging (LiDAR) based elevation mappings on a regular basis in Finland. This data is openly available for anyone to download [2] and contains laser scanned point-cloud data with approximately one point per 2 m^2 [9]. The LiDAR data set was acquired for each of the four fields. The LiDAR data were converted from point-cloud data to spatial rasters using the ArcGIS (Esri, Redlands, California, USA) software. During the conversion, the data were interpolated to match UAV data data in terms of resolution. The topography data contains only the following layer:

37. Elevation information

2.1.6 Weather data

Weather data were collected with two separately located Vantage Pro2 (Davis Instruments, Hayward, California, USA) weather stations. As the fields constitute two distinct clusters, a weather station was placed in the immediate

vicinity of each field cluster. While the stations log multiple variables with a time resolution of just minutes, we utilized accumulated daily statistics and matched data to UAV acquisition dates. Thus, five weather data maps were constructed for each field spacing matching the dates of the UAV data. The weather data contains the following layers:

38. Cumulative temperature sum
39. Cumulative rain sum

2.1.7 Yield data

As the task of regression is that of supervised prediction, the training of the CNN model requires information about the ground truth, the target values. These were acquired during the harvest of 2018 via yield mapping sensor devices attached to the harvesters, either with a CFX 750 (Trimble Navigation, Sunnyvale, California, USA) or Greenstar 1 (John Deere, Molinde, Illinois, USA). CFX 750 utilizes optical sensors to measure yield throughput and moisture. Greenstar 1 utilizes a kinetic mass flow sensor to measure yield throughput and a separate moisture sensor. The yield maps generated by the mapping equipment were initially in the form of vector point-clouds. The irregularly spaced points were filtered prior rasterization to contain only points where the yield was between 1500 and 15000 kg/ha and the harvester speed between 2 and 7 km/h, following the yield pre-processing methodology of [11]. Rasterization was then done by interpolating the yield data to form a raster image.

2.2 Data Preprocessing

2.2.1 Interpolation

The first step after the acquisition of data was to harmonize the spatial resolution across multiple different sources. The UAV data were initially downsampled to 0.3125 m/px, or 32 pixels per 10 meters. This is to match the method of data processing in [11]. Main reasons are to limit the inputs to reasonable size and to have the input dimensions conform to a power of 2 for GPU-based computations. The coarser data, namely Sentinel-S2, soil samples, Veris MSP3, elevation and yield data, required upsampling via interpolation to match this resolution. The interpolation was done using the GDAL utility's `gdal_grid` program with `invdist:power=3:smoothing=20` interpolation algorithm. As with the input data, also the target crop yield data were interpolated to UAV matching resolution. Example results of interpolation are depicted in Figure 1.

2.2.2 Input Feature normalization

After interpolation, the next step was to normalize the data. While absolute values could also be directly used, scaling the input values close to the magni-

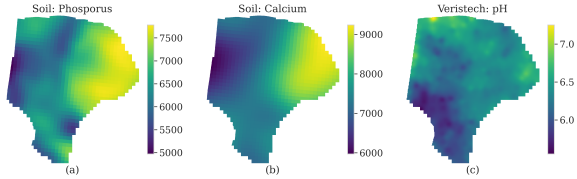

Fig. 1 Examples of input data interpolations on field-scale. (*a*) is the interpolated phosporus map, (*b*) the interpolated calcium content in the field and (*c*) the pH map as measured by the Veris MSP3 soil mapper.

tude of the model's parameters (i.e. weights) helps the model converge faster. Input layers were normalized using a function

$$d^{NORM} = (d - \mu_d)/(max(d) - min(d)), \quad d \in D \tag{1}$$

where d is a layer in the set of all layers D in the data set and d^{NORM} is the normalized layer. However, the target crop yield values were not scaled, akin to [11].

2.2.3 Frame separation

The CNNs require input data to have fixed dimensions. Low number of fields and the irregularities of field shapes led us to extract smaller, fixed dimension frames from the field data. Following [11], we extracted overlapping 40 $\times$ 40 m (128 $\times$ 128 px) frames with 10 m horizontal and vertical steps. Prior extraction, all input and target data from various input sources were aligned in terms of geolocation and resolution to ensure frame extraction from mathching areas. Frames containing half or more valid pixels were included in data, while those having less than half were discarded. This resulted in a total of 16375 input-target frames.

2.2.4 Data sets

Extracted samples were then divided into training, validation and test sets. Training and validation sets were utilized during the training, while the test set was set aside as the out-of-sample performance evaluation data set. As the number of unique fields was low, we wanted to maximize the sample variability the model sees during training. We first attempted to train the models with data separated on a per-field basis with two fields for training, one for validation and one for testing. This led to the model overfitting to the training data and poor generalization performance due to low training data set variability. Similarly low performance was achieved with splitting fields to separate

Table 3 Compositions of training, validation and test sets used to train and evaluate the models.

Data set	Weeks	Frames	Proportion
Training	21,23,25	7561	46.2%
Validation	24	2938	17.9%
Test	22, 26	5876	35.9%

training, validation and test sections. We thus decided then to divide the data temporally to distinct training, validation and test sets according to the UAV image acquisition week. The samples were then shuffled to eliminate spatial autocorrelation in subsequent samples due to overlapping frame extraction. Used weeks, sample counts and sample count proportions for separated sets are given in Table 3.

2.3 Model Architecture

Convolutional neural networks, CNNs, are a subset of spatial model architectures within the broader context of deep learning. CNNs excel in tasks, where the inputs fed to the model are either images or image-like data, i.e. spatial data [14, 7]. While the inner workings of the CNNs has already been well documented [11], we quickly review the operating principles of a CNN. The architecture operates with layers, like many of the deep learning architectures. Each layer is a combination of a convolution operation, which is often followed by a pooling operation. At the heart of the model are the trainable filters of the convolution operation, i.e. the kernels, which produce feature maps for further use.

In our study, we implement and use the best performing CNN architecture of [11]. The model consists of six convolutional layers, followed by two fully connected (FC) layers. Convolutional layers consist of 2D convolutions, batch normalization and non-linear activation with a rectified linear unit (ReLU). First and last convolutional layers also employ max pooling with 2×2 kernel to extract more robust features and reduce intermediate output data dimensions. First five convolutional layers operate with 64 5×5 kernels and the last convolutional layer with 128 5×5 kernels. The outputs of the last convolutional layer are then flattened to a single vector, which is then fed to two 1024 neuron FC layers, both having ReLU activation. Last FC layer outputs the final prediction result. The model was implemented with PyTorch [12] and trained with Skorch [16].

2.4 Training

To gauge the effects of multisource data on the crop yield prediction task with spatial inputs, we performed trainings with four different input data configurations. The data configurations and the input data sources included in

Table 4 The different data configurations used for training distinct models. *RGB Only* uses UAV RGB data only. *No S2* uses UAV, soil, Veris MSP3, topography and weather data. *S2 Raw* adds Sentinel-S2 raw wavelength band data to *No S2*. *S2 Full* adds calculated Sentinel-S2 Level-2A product layers to *S2 Raw*. An X indicates the inclusion of an input data source to a data configuration, while a dash indicates the exclusion.

Source	Channels	RGB Only	No S2	S2 Raw	S2 Full
UAV	1-3	X	X	X	X
Soil	23-30	-	X	X	X
Veris	31-36	-	X	X	X
Topo	37	-	X	X	X
Weather	38-39	-	X	X	X
S2 bands	4-15	-	-	X	X
S2 other	16-22	-	-	-	X
Band count		3	20	32	39

them are further given in Table 4. To elaborate, the derived data configurations were as follows:

- *RGB only.* As [11] was conducted with RGB data from UAVs only, we wanted to make baseline performance evaluation with UAV RGB data only. No other sources were included in this setting.
- *No S2.* Next we wanted to see the effects of soil and weather data on the predictive performance. We thus included all other sources of data (UAV, soil, Veris MSP3, topography and weather) but excluded the satellite data.
- *S2 Raw.* As Sentinel-S2 Level-2A products contain additional algorithmically generated layers, we wanted to see the effect of including just the raw wavelength bands with other input data sources.
- *S2 Full.* The last setting was to use all data acquired for this study.

Because data were distinct from data used in [11], we initialized and trained all models anew for each data configuration. To account for the effects of randomized network parameter initialization, we trained 10 models per data configuration, 40 trainings in total. We used Adadelta [19] as the optimizer, 0.58 for the learning rate, 0.001 for the weight decay and 0.9 for the Adadelta's ρ coefficient as those were the best performing hyperparameters in [11]. Similarly, we used early stopping with a patience of 50 stagnant epochs and continued the training once. The models were trained with Nvidia Tesla V100 Volta and Pascal architecture server GPUs in a distributed computation environment.

3 Results

The CNN models with distinct input data configurations were trained with data of four unique fields. The model architectures, hyperparameters and the training procedures were identical to [11]. As the aim of our study was to evaluate the effects of introducing multisource inputs to crop yield prediction, we trained spatial yield prediction models with four distinct data configurations. The data configurations are discussed in Section 2.4. As the training time loss

Table 5 The test set performance of the same CNN architecture and hyperparameter configuration with various data configurations. *RGB Only* is the baseline model. Out of the configurations, the model performed best with all input data layers (*S2 Full*).

Data Configuration	**Test RMSE (kg/ha)**	**Test MAE (kg/ha)**	**Test MAPE (%)**	**Test R^2 -**
RGB Only	1055.7	838.8	18.2	0.343
No S2	892.4	694.9	14.8	0.531
S2 Raw	461.0	340.9	6.94	0.875
S2 Full	**364.1**	**274.3**	**5.18**	**0.922**

function we used the mean squared error (MSE). Other loss metrics were also calculated, including the square root of the MSE (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE) and the coefficient of determination (R^2). These metrics were not, however, monitored and neither did they not influence model selection during training.

The baseline model using UAV RGB data only attained 1055.7 kg/ha test RMSE, 18.2% test MAPE and 0.343 test R^2. Out of all data configurations, the best performance of 364.1 kg/ha test RMSE, 5.18% test MAPE and 0.922 test R^2 was achieved using all input data presented in our study (*S2 Full*). The performance results for all data configurations with the held out test data set are given in Table 5.

To gain a better view into how the models train with distinct data predicted, we also examined the unseen test sample distributions of predicted values against ground truth values, the true crop yields. With the data, the baseline *RGB Only* model's predictions resemble a Gaussian distribution centered around the mean 5140 kg/ha of true yield values. As more inputs are introduced, the predicted distributions' shapes align with the true values more closely, expressing multi-modal peaks where the true values have them. The test set distributions are depicted in Figure 2.

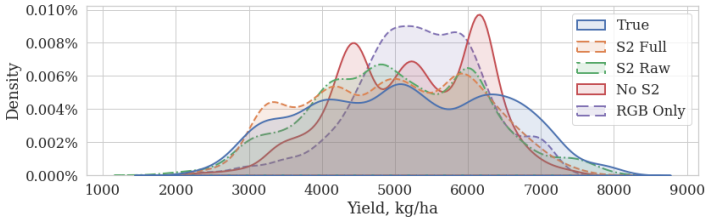

Fig. 2 Distributions of predictions against true yields with the holdout test set.

Table 6 The relative performance of the models trained with distinct multisource input data configurations to the baseline *RGB Only* model. For RMSE, MAE and MAPE negatiThe input data configurations are defined in Section 2.4.

Data Setting	Relative change from *RGB Only*			
	Test RMSE	**Test MAE**	**Test MAPE**	**Test R^2**
No S2	-15.5%	-17.2%	-18.7%	+0.188
S2 Raw	-56.3%	-59.4%	-61.9%	+0.532
S2 Full	**-65.6%**	**-67.3%**	**-71.5%**	**+0.579**

4 Discussion and Conclusions

In this study, we evaluated the effects of using input data from multiple sources on the task of spatial crop yield prediction. Using a CNN model architecture developed for UAV RGB inputs from [11], we introduced additional data from sources like soil samplings, Veris MSP3 soil scanner, topographical maps, weather stations and Sentinel-S2 satellites to the model. We trained ten models for each distinct input data configuration: (1) a *RGB Only* baseline model, (2) a *No S2* multisource model with satellite data excluded, (3) a *S2 Raw* multisource model with raw satellite band data included and (4) a *S2 Full* multisource model with all input data. Out of each set of ten trained models, we selected the models performing best. The model architecture and hyperparameters for the training were taken from [11] and left unchanged to constrain the variability in performance to data only. Only thing varying between model trainings, in addition to four distint input data source configurations, were the initialized model weights.

The performance with larger number of fields using UAV RGB data has already been extensively studied in our previous studies [11] and [10]. Thus, training a model with only UAV RGB data provides a studied baseline to which models trained with additional data can be compared against. The best performing data configuration was *S2 Full* with 364.1 kg/ha test RMSE, 5.18% test MAPE and 0.922 test R^2 using all 39 layers of input data for each extracted frame. Compared to the baseline *RGB Only* model, the *S2 Full* attained 65.6% lower RMSE, 67.3% lower MAE, 71.5% better MAPE and 0.579 higher R^2 with the test set. Generally every model with multisource inputs performed better than the baseline model. This is shown in Table 6.

Crop yield prediction with spatial data and spatial deep learning models has seen an increase in the past few years [6]. Having been studied with a variety of different architectures, from feed-forward networks to hybrid spatio-temporal models, studies have also been conducted with CNN as the main architecture. In [11], a single CNN model was developed to predict crop yields from fields with varying crop types (wheat, barley and oat) from UAV images collected of Finnish crop yields during 2017. Using smaller frames extracted from ortho-images, the best performance was 484.3 kg/ha MAE and 8.8% MAPE. Using soil nutrient data, seed rate, elevation maps, soil's electroconductivity and satellite data in USA, [4] trained a CNN to predict crop yields for nine fields. They report an average scaled MSE of 0.70 which translates to

1145 kg/ha. [18] utilized RGB and multispectral data acquired with a UAV from rice fields in China to predict rice yields with a composite CNN model on field block scale. Feeding the multisource data to distinct, parallelized CNNs, they report a rice yield prediction performance of 0.50 R^2 and 26.6% MAPE.

As we had sufficient data overlap across multiple input sources and the data were acquired from only four unique fields, objective multisource crop yield prediction performance evaluation requires more care in interpreting the results. Relative increase in performance from best performing UAV data utilizing *RGB only* model to the best *No S2* model with additional soil and weather data was notably small. Largest improvements were gained with the introduction of Sentinel-S2 data. Adding raw Sentinel-S2 bands to the RGB, soil and weather data increased the performance by 40.8% RMSE, 42.2% MAE, 43.2% MAPE and 0.344 R^2 from *No S2*. Thus, the increase in performance with Sentinel-S2 is considerably higher than what was achieved with adding soil, topography and weather data to UAV RGB data.

Data acquisition for remote sensing and multisource input data for smart farming is generally laborous and resource intensive. While satellite data is generated automatically, UAVs require semi-autonomous operation at best and the collection of soil data requires extensive on-site manual labour. With more data from a variety of sources a more extensive and representative study can be conducted.

Another limitation stems from differences in spatial and temporal dispersion of different input data sources. UAV, Sentinel-S2 and weather data vary temporally in the data we have used, whereas soil samplings, Veris MSP3 and topographical maps do not. As our data was split temporally to training, validation and test sets, the latter are present in all of these data sets. On the other hand, weather data varies only temporally and constitutes spatial rasters with constant values corresponding to the time of UAV imaging. This means that whether the data is split temporally or spatially, some layer or part of data is always present in training, validation and test sets. As [13] point out, deep learning models are able to implicitly learn linear and non-linear couplings from data with correlations. This means that the deep learning models learn sets of representative features from complex combinations of the inputs and not from single input values on solitude. Furthermore, the performance gains with UAV RGB data combined with temporally invariant soil and ground data is trumped by the performance gains of data configurations using Sentinel-S2 data as additional inputs. This would suggest that the combination of the inputs matters more than presence of distinct, invariant data in training, validation and test sets. However, the concrete effects of simultaneous layer-level data existence in training, validation and test data sets are presently unknown to us and, thus, a subject of future research.

Regarding multisource data in the context of smart farming and crop yield estimation, data itself is an evolving research topic. The use of multisource inputs in remote sensing, while focusing on multispectral data acquired from satellite systems orbiting the globe, has been extensively reviewed in [5]. The use of multispectral data from UAVs and the prediction architectures thereof

is also a developing topic [8]. Another topic related to spatial data is that of autocorrelation [3]. To address autocorrelation of spatial frames in a future study, the inclusion of pixel-wise location information, as suggested in [3], should be sufficient to inform the deep learning model whether data similarity is due to proximity or some other factor or combination of them.

In conclusion, our study indicates that increasing the number of input data sources increases the performance of intra-field crop yield prediction. To draw definite conclusions on the most optimal configuration of input data sources more data is required. With more representative data, generalizable conclusions are more warranted. As the data in this study focuses on a single rowing season, a future plan is to study the generalization of a multisource crop yield prediction model with multiple years of data. Yet in this study the relative increase from baseline of using UAV RGB only as the input data were notable. Consolidating UAV RGB data with soil and ground topology data already somewhat improves the prediction performance, while largest performance gains were gained from using Sentinel-S2 in addition to UAV RGB, soil sampling, Veris MSP3 soil scanner, weather and topography data.

Acknowledgements We would like to thank Mtech Digital Solutions Oy for partially funding this research, Tampere University for providing the computational resources and MIKÄ DATA project for providing us with data.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. ESA: Sentinel-2. URL https://sentinel.esa.int/web/sentinel/missions/sentinel-2
2. PaITuli - Spatial data for research and teaching. URL https://paituli.csc.fi/download.html
3. Amgalan, A., Mujica-Parodi, L.R., Skiena, S.S.: Fast Spatial Autocorrelation. Biometrics **30**(4), 729 (2020). DOI 10.2307/2529248. URL http://arxiv.org/abs/2010.08676 https://www.jstor.org/stable/2529248?origin=crossref
4. Barbosa, A., Marinho, T., Martin, N., Hovakimyan, N.: Multi-stream CNN for spatial resource allocation: A crop management application. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, vol. 2020-June, pp. 258–266 (2020). DOI 10.1109/CVPRW50498.2020.00037
5. Ghamisi, P., Rasti, B., Yokoya, N., Wang, Q., Hofle, B., Bruzzone, L., Bovolo, F., Chi, M., Anders, K., Gloaguen, R., Atkinson, P.M., Benediktsson, J.A.: Multisource and multitemporal data fusion in remote sensing: A comprehensive review of the state of the art. IEEE Geoscience and Remote Sensing Magazine **7**(1), 6–39 (2019). DOI 10.1109/MGRS.2018.2890023
6. van Klompenburg, T., Kassahun, A., Catal, C.: Crop yield prediction using machine learning: A systematic literature review. Computers and Electronics in Agriculture **177**, 105709 (2020). DOI 10.1016/j.compag.2020.105709. URL https://linkinghub.elsevier.com/retrieve/pii/S0168169920302301
7. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Communications of the ACM **60**(6), 84–90 (2017). DOI 10.1145/3065386. URL http://dl.acm.org/citation.cfm?doid=3098997.3065386

8. Messina, G., Modica, G.: Applications of UAV thermal imagery in precision agriculture: State of the art and future research outlook. Remote Sensing **12**(9) (2020). DOI 10.3390/RS12091491
9. National Land Survey of Finland: Laser scanning data. URL http://www.nic.funet.fi/index/geodata/mml/laserkeilaus/mml_laserkeilaus_2016_eng.pdf
10. Nevavuori, P., Narra, N., Linna, P., Lipping, T.: Crop Yield Prediction Using Multitemporal UAV Data and Spatio-Temporal Deep Learning Models. Remote Sensing **12**(23), 4000 (2020). DOI 10.3390/rs12234000. URL https://www.mdpi.com/2072-4292/12/23/4000
11. Nevavuori, P., Narra, N., Lipping, T.: Crop yield prediction with deep convolutional neural networks. Computers and Electronics in Agriculture **163**(June), 104859 (2019). DOI 10.1016/j.compag.2019.104859. URL https://linkinghub.elsevier.com/retrieve/pii/S0168169919306842
12. Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., Lerer, A.: Automatic differentiation in PyTorch. In: NIPS-W (2017)
13. Sun, Y., Guo, G., He, X., Liu, X.: Multi-level coupling network for Non-IID sequential recommendation. IEEE Access **7**(Iid), 186247–186259 (2019). DOI 10.1109/ACCESS.2019.2961182
14. Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A.: Going deeper with convolutions. Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition **07-12-June**, 1–9 (2015). DOI 10.1109/CVPR.2015.7298594
15. Tedesco-Oliveira, D., Pereira da Silva, R., Maldonado, W., Zerbato, C.: Convolutional neural networks in predicting cotton yield from images of commercial fields. Computers and Electronics in Agriculture **171**, 105307 (2020). DOI 10.1016/j.compag.2020.105307. URL https://linkinghub.elsevier.com/retrieve/pii/S0168169919319878
16. Tietz, M., Fan, T.J., Nouri, D., Bossan, B., skorch Developers: skorch: A scikit-learn compatible neural network library that wraps PyTorch (2017). URL https://skorch.readthedocs.io/en/stable/
17. Warmerdam, F., Rouault, E.: GDAL — GDAL documentation (1998). URL https://gdal.org/
18. Yang, Q., Shi, L., Han, J., Zha, Y., Zhu, P.: Deep convolutional neural networks for rice grain yield estimation at the ripening stage using UAV-based remotely sensed images. Field Crops Research **235**(August 2018), 142–153 (2019). DOI 10.1016/j.fcr.2019.02.022. URL https://doi.org/10.1016/j.fcr.2019.02.022 https://linkinghub.elsevier.com/retrieve/pii/S037842901831390X
19. Zeiler, M.D.: ADADELTA: An Adaptive Learning Rate Method (2012). DOI http://doi.acm.org.ezproxy.lib.ucf.edu/10.1145/1830483.1830503. URL https://arxiv.org/pdf/1212.5701.pdf http://arxiv.org/abs/1212.5701