

Deep Hybrid Neural Networks on Multi-temporal Satellite Data: Predicting Farm-scale Crop Yields

MARTIN ENGEN,
ERIK SANDØ,
BENJAMIN LUCAS OSCAR SJØLANDER

SUPERVISORS

Simon Arenberg (inFuture AS)
Morten Goodwin (University of Agder)

University of Agder, 2021
Faculty of Engineering and Science
Department of ICT



Abstract

Accurate farm-scale crop yield predictions can enable farmers to improve their yield per decare and inform subsequent sectors of the availability of grains sooner. Existing research on yield predictions is limited to regional analytics, which often fails to capture local yield variations influenced by farm management decisions and field conditions. Farm-scale crop yield predictions require precise ground-truth prediction targets, which are not always available. It takes substantial manual labor to create large and suitable datasets of high-resolution per-farm samples.

This thesis introduces a hybrid multi-temporal deep neural network that combines convolutional and recurrent features specially designed to predict the individual crop yields of farms across Norway with per-farm samples. To the best of our knowledge, this is the first farm-scale crop yield prediction model of its kind. The hybrid model learns to extract features from both multi-temporal satellite images and weather data time series to predict crop yields accurately. We use a complex multitude of noisy data sources, including multi-temporal satellite images from Sentinel-2, weather data from The Norwegian Meteorological Institute, farm data and grain delivery data from the Norwegian Agriculture Agency, and cadastral data.

Our hybrid model, which combines two and one-dimensional convolutional layers and a gated recurrent unit network, predicts crop yields with an error of 76 kg/daa using satellite images and weather data, according to our experiments.

Contents

| | |
|-------------------------------------------------------------------------------------------|-----------|
| Abstract | i |
| Glossary | v |
| 1 Introduction | 1 |
| 1.1 Background | 1 |
| 1.1.1 KORNMO | 1 |
| 1.1.2 Preliminary Project | 2 |
| 1.2 Problem Statement and Hypotheses | 2 |
| 1.2.1 Hypotheses | 2 |
| 1.3 Thesis Outline | 3 |
| 2 Theoretical Background | 5 |
| 2.1 Artificial Neural Networks | 5 |
| 2.1.1 The Perceptron | 5 |
| 2.1.2 Deep Neural Networks | 6 |
| 2.1.3 Convolutional Neural Networks | 7 |
| 2.1.4 Recurrent Neural Networks | 8 |
| 2.2 Plant Growth Factors | 10 |
| 2.2.1 Light | 10 |
| 2.2.2 Temperature | 10 |
| 2.2.3 Water | 10 |
| 2.2.4 Nutrition | 11 |
| 2.3 Remote Sensing | 11 |
| 2.3.1 Vegetation index | 11 |
| 3 State-of-the-art | 14 |
| 3.1 Origins and Early Use of Remote Spectral Observations | 14 |
| 3.1.1 A Large Area Crop Inventory Experiment | 15 |
| 3.1.2 Using Remote Sensed Information as Input for Agrometeorological Models | 15 |
| 3.1.3 Summary | 16 |
| 3.2 Deriving Value from Vegetation Indices | 16 |
| 3.2.1 NDVI to Estimate Wheat Yield in Italy 1986-1989 | 16 |
| 3.2.2 Relationship of NDVI and Weather Features to Corn and Soybean Yields | 16 |
| 3.2.3 Summary | 17 |
| 3.3 Machine Learning Applied to Remotely Sensed Data | 17 |
| 3.3.1 Growth Phases as Timesteps in LSTM | 17 |
| 3.3.2 Satellite Image Pixel Histograms as Input to CNN and LSTM | 18 |

| | | |
|----------|----------------------------------------------------------------------------|-----------|
| 3.3.3 | Using Raw Satellite Images to Predict Wheat Yield in India | 18 |
| 3.3.4 | Field-scale Yield Prediction of Corn and Soy | 19 |
| 3.3.5 | Summary | 19 |
| 4 | Multi-Temporal Data and Data Handling | 21 |
| 4.1 | Norwegian Agriculture Agency | 21 |
| 4.1.1 | Grain Deliveries | 21 |
| 4.1.2 | Production Subsidies | 22 |
| 4.1.3 | Land Use | 22 |
| 4.2 | Geographical Data | 23 |
| 4.2.1 | Cadastral Data | 23 |
| 4.2.2 | Field Boundaries | 23 |
| 4.2.3 | Combined Cadastral and Field Data | 25 |
| 4.3 | Weather Data | 25 |
| 4.3.1 | Collecting Weather Features | 25 |
| 4.3.2 | Interpolation | 26 |
| 4.4 | Satellite Data | 27 |
| 4.4.1 | Sentinel-2 | 27 |
| 4.4.2 | Building a Multi-Temporal Image Dataset | 28 |
| 4.5 | Masking | 28 |
| 4.5.1 | Generating Masks | 31 |
| 4.5.2 | Applying Masks | 31 |
| 4.6 | Time Spans of Satellite and Weather Data | 32 |
| 4.7 | Summary | 32 |
| 5 | Method | 34 |
| 5.1 | Data Preprocessing | 34 |
| 5.1.1 | Normalization | 34 |
| 5.1.2 | Prediction Targets | 35 |
| 5.2 | Weather Data | 36 |
| 5.2.1 | The Weather DNN Model | 36 |
| 5.3 | Satellite Images | 36 |
| 5.3.1 | The Single Image CNN Model | 37 |
| 5.3.2 | A Multi-Temporal CNN-RNN Model | 37 |
| 5.4 | Satellite Images and Weather | 38 |
| 5.4.1 | Handcrafted Features in LSTM | 38 |
| 5.4.2 | Hybrid 1: Pre-Trained Hybrid Model | 40 |
| 5.4.3 | Hybrid 2: Hybrid CNN Model | 41 |
| 5.5 | Reducing Overfitting | 42 |
| 5.5.1 | Data Augmentation | 42 |
| 5.5.2 | Stochastic Epoch Sampling | 44 |
| 6 | Experiments and Results | 46 |
| 6.1 | Initial Experiments on Multispectral Images | 46 |
| 6.1.1 | Evaluating the Use of Multispectral Imagery for Yield Prediction | 46 |
| 6.1.2 | Optimal Week to Predict Yield | 47 |
| 6.1.3 | Effects of Data Augmentation | 49 |
| 6.1.4 | Effects of Masking | 52 |
| 6.2 | Crop Yield Model Comparisons | 53 |
| 6.2.1 | Multi-Temporal CNN-RNN | 53 |
| 6.2.2 | Handcrafted Features in LSTM | 54 |

| | |
|-------------------------------------------------------|-----------|
| 6.2.3 Hybrid 1: Pre-Trained Hybrid | 54 |
| 6.2.4 Hybrid 2: Hybrid CNN | 55 |
| 6.3 Early Predictions | 59 |
| 6.4 Predictions as Regional Analytics | 62 |
| 6.5 Summary | 63 |
| 7 Conclusions | 64 |
| 8 Future Work | 66 |
| 8.1 Improving Generalization | 66 |
| 8.2 Remote Sensed Temperature | 66 |
| 8.3 NIBIO Field Data | 66 |
| 8.4 Additional Sources for Satellite Images | 66 |
| Bibliography | 68 |

Glossary

API Application Programming Interface. 23, 25, 27, 28

cadastral Of or relating to the cadastre. 3, 21–23, 25, 31

cadastre A comprehensive land recording of real estate of a country. 21, 23, 25

CNN Convolutional Neural Network. 7, 8, 18, 19, 36–38, 40–42, 46, 47, 53–56, 59, 63

Copernicus The European Union’s Earth observation program. 1

decare Surface area unit — Equivalent to 1000 m² or 0.1 hectare. i, 22, 34, 35, 46, 53

DNN Deep Neural Network. 6, 18, 26, 27, 38, 40, 41, 53–55, 63

GRU Gate Recurrent Unit, a type of recurrent neural network. vii, 9, 10, 38, 42, 54

kg/daa Kilograms per Decare — The area-adjusted unit for crop yield used in this thesis. i, 22, 27, 34, 35, 37, 39, 46, 53–56, 59, 62, 64

LAI Leaf Area Index. 16

LST Land Surface Temperature. 16, 17

LSTM Long-Short-Term Memory, a type of recurrent neural network. vii, 9, 10, 17–19, 38–40, 54

MAE Mean Absolute Error. 46, 54, 62

MLP Multilayer Perceptron. 6

MSI Multispectral Instrument — The Sentinel-2 satellites imaging instrument, capturing light in 13 spectral bands. 27

MSS Multispectral Scanner — The imaging instrument on the Landsat 1 satellite. 14, 16

NDVI Normalized Difference Vegetation Index. 11, 12, 16–18

NIBIO Norwegian Institute for Bioeconomy Research. 23, 25, 31, 66

NIR Near Infrared. 12, 27

NN Neural Network. 9

phenology "[The] study of the timing of recurring biological events" [41]. 17, 37

ReLU Rectified Linear Unit — A common activation function. 37, 38, 40–42

RMSE Root Mean Square Error. 17, 18, 62

RNN Recurrent Neural Network. vii, 8–10, 38, 40, 41, 53–55

Sigmoid A common activation function. 9

SWIR Short-Wave Infrared. 11, 18, 27

tanh Hyperbolic tangent activation function – Activation function used in neural networks. 9, 36, 40

TLU Threshold Logic Unit. 5

VNIR Visible and Near Infrared. 11, 18

List of Figures

| | | |
|------|--------------------------------------------------------------------------------|----|
| 2.1 | The threshold logic unit | 6 |
| 2.2 | A Multilayer Perceptron | 6 |
| 2.3 | Convolutional Layers | 7 |
| 2.4 | Illustration of a Recurrent Neural Network | 8 |
| 2.5 | Illustrations of a simple RNN cell, an LSTM cell, and a GRU cell. | 9 |
| 2.6 | The spectral reflectance curves of different earth surfaces | 12 |
| 2.7 | NDVI applied to satellite image of a farm | 13 |
| 4.1 | Visualization of the geographical layers, overlaid on a satellite map. | 24 |
| 4.2 | The neural network architecture used for precipitation interpolation | 26 |
| 4.3 | The neural network architecture used for temperature interpolation | 26 |
| 4.4 | Image time series for a farm | 30 |
| 4.5 | From cultivated fields to generated mask | 31 |
| 4.6 | Mask applied to an image | 32 |
| 4.7 | Process of applying mask | 32 |
| 4.8 | Process of adding mask | 33 |
| 5.1 | Prediction model system architecture | 35 |
| 5.2 | Weather DNN architecture | 37 |
| 5.3 | Single Image CNN Model architecture | 38 |
| 5.4 | Multi-Temporal CNN-RNN Model architecture | 39 |
| 5.5 | Handcrafted features in LSTM architecture | 40 |
| 5.6 | Pre-trained Hybrid Model architecture | 41 |
| 5.7 | Hybrid CNN Model architecture | 42 |
| 5.8 | Cropping augmentation visualized in true color | 43 |
| 5.9 | Rotation augmentation visualized in true color | 44 |
| 5.10 | Salt-and-pepper augmentation visualized in true color | 44 |
| 5.11 | Illustration of stochastic epoch sampling | 45 |
| 6.1 | The initial results of the Single Image CNN model | 47 |
| 6.2 | Weekly performance comparison of the Single Image CNN model | 48 |
| 6.3 | Training and validation loss of cropping and rotating | 50 |
| 6.4 | Training and validation loss of applied salt-and-pepper noise | 50 |
| 6.5 | The effects of augmentation methods on MAE | 51 |
| 6.6 | The effects of masking techniques on MAE | 52 |
| 6.7 | Training and validation loss for the Multi-temporal CNN-RNN model | 54 |
| 6.8 | Training and validation loss for the LSTM model | 55 |
| 6.9 | Training and validation loss for the Pre-Trained Hybrid model | 55 |
| 6.10 | Training and validation loss for the Hybrid CNN | 56 |
| 6.11 | Hybrid CNN prediction quantiles versus real quantiles | 57 |

| | | |
|------|----------------------------------------------------------------------------|----|
| 6.12 | Discretized prediction output from the Hybrid CNN | 57 |
| 6.13 | Discretized prediction output from the Hybrid CNN in percentile bins . . . | 58 |
| 6.14 | Early Predictions: Late-June | 60 |
| 6.15 | Early Predictions: Mid-May | 61 |
| 6.16 | Relationship between actual and predicted crop yields on a commune-scale | 62 |

List of Tables

| | | |
|-----|------------------------------------------------------------|----|
| 2.1 | Sum day degrees | 10 |
| 4.1 | MAE in weather interpolation | 27 |
| 4.2 | Change in MAE when using interpolated weather | 27 |
| 4.3 | Sentinel-2 Spectral Bands | 29 |
| 4.4 | Farm coverage for different bounding box sizes | 30 |
| 5.1 | Feature normalization constants | 35 |
| 5.2 | Handcrafted features from satellite images | 40 |
| 6.1 | Effects of augmentations | 49 |
| 6.2 | Best mean absolute error achieved for each model | 53 |
| 6.3 | Hybrid CNN results with and without masks | 56 |
| 6.4 | MAE of early predictions | 59 |

Chapter 1

Introduction

Yield prediction using machine learning is an increasingly researched topic worldwide and has been applied in agricultural use for some time [38]. Knowing when and how much of a particular crop will be produced can be an important tool for improving food security and aiding decision-making at various administrative levels. The development of remote sensing data from satellite sensors has allowed for easy access to vast datasets on a global scale, lessening the need for manual and locally collected data, which is often challenging to scale and ineffective [42].

While crop yield prediction studies have previously studied yield prediction models for regional or national scales, few published studies have been performed on field-scale or farm-scale yield prediction [34]. The lack of farm-scale yield predictions has been attributed to a lack of "ground-truth" data for crop yield targets (e.g. kg per decare) on a per-farm basis[34], as well as a lack of funding and the high cost of collecting satellites images [34][21]. However, these impediments seem to be fading: in Norway, detailed agricultural reports, including per farm statistics, have been made publicly available since 2017, and high-resolution satellite images are obtainable through the European Union's Earth observation program Copernicus.

With the increased availability of data on a per-farm basis coupled with satellite images from the Copernicus Sentinel-2 mission, the main motivation behind this thesis is to explore the use of satellite imagery and deep learning to predict grain yields on a per-farm basis throughout Norway.

1.1 Background

1.1.1 KORNMO

The KORNMO project is a collaborative research effort between Felleskjøpet, University of Agder, InFuture, Microsoft, and the Confederation of Norwegian Enterprise (NHO). The goal of KORNMO is to increase the quality, efficiency, and sustainability of Norwegian grain production through the use of machine learning applied to agriculture. Some of the specific user scenarios are to give benchmarks and optimization advice to the farmers, and quality management assistance at mills where the grains are delivered and stored.

One of the objectives of KORNMO is accurate crop yield predictions. Having accurate crop yield predictions could both serve as a benchmark to the individual farmers and

assist in making estimates to the mills of how much grains will be delivered in a season. As such, crop yield prediction was proposed as a relevant area of study, which led to a preliminary project exploring the feasibility of predicting grain crop yields across Norway.

1.1.2 Preliminary Project

The preliminary project[36] looked at yield prediction on Norwegian grain farms using an artificial neural network on weather data and information on each farm. While experiments showed that weather data is useful in predicting crop yields, there are still unexplained differences in farmers' crop yield. This thesis builds on the results from the preliminary project by adding additional data sources and more complex neural networks, most notably satellite data and convolutional, recurrent, and hybrid networks.

1.2 Problem Statement and Hypotheses

Given that weather data is useful to farm-scale yield prediction and satellite data is useful to regional yield prediction, this thesis explores if and how neural networks can use satellite data for farm-scale yield prediction, and if models that use both surpass the predictive abilities of simpler models. To further concretize the problem, we define four hypotheses which this thesis will test.

1.2.1 Hypotheses

Hypothesis 1: *Satellite images of farms and their surroundings can be used to accurately predict farm-scale crop yields.*

The first hypothesis assumes that farm-scale crop yield prediction is possible given the availability of enough per-farm data in Norway and satellite images, and that results are at least comparable to the results from the preliminary project using weather data. This is a prerequisite for the subsequent hypotheses, which all assume that satellite data contains some independent variables that affect crop yield.

Hypothesis 2: *Accurate field boundaries along with satellite images increase crop yield accuracy significantly.*

This hypothesis assumes that differences between field conditions and management decisions in neighbouring farms can affect crop yield, which could be difficult for a model to learn unless accurate field boundaries are provided. If the hypothesis is correct, it may indicate that satellite images can be used to explain differences in crop yield between neighbouring farms, and that such models can be used to aid in decision-making at a farm level.

Hypothesis 3: *Prediction accuracy can be further increased by combining satellite images and weather data.*

It is assumed that weather data and satellite images contain some different and independent variables. We hypothesize that a deep learning model is able to learn features

from both datasets effectively and that this provides better performance compared to models using the two data sources separately. Higher prediction accuracy makes the prediction model more useful for aiding farmers and subsequent industries.

Hypothesis 4: *It is possible to predict farm-scale crop yield earlier in the growing season with some reduced accuracy.*

The most accurate crop yield predicts will likely be when there is as much data available as possible, meaning at the end of the growing season. However, getting accurate estimates for deliveries earlier allows mills and administrative authorities to prepare in advance.

1.3 Thesis Outline

Chapter 2 lays the theoretical foundation behind the topics and work covered in the thesis. Brief overviews are given for 2.1 Artificial Neural Networks, 2.2 Plant Growth Factors, and 2.3 Remote Sensing.

Chapter 3 investigates how the use of remotely spectral observations regarding vegetation and agriculture has progressed over time and defines the current state-of-the-art for yield predictions using remotely sensed data. 3.1 Origins and Early Use of Remote Spectral Observation investigate the original purpose of earth-observing satellites. 3.2 Deriving Value from Vegetation Indices explores how handcrafted features derived from remotely sensed data are used. 3.3 Machine Learning Applied to Remotely Sensed Data explores how modern machine learning techniques have been used to predict yields using remotely sensed data in the last few years.

Chapter 4 explains the multi-temporal data sources used and how these are handled. 4.1 describes the data published yearly from the Norwegian Agriculture Agency. 4.2 Geographical Data shows how farms are mapped to geographical locations and how field boundaries are gathered from the soil quality dataset. 4.3 Weather Data describes how the weather features are downloaded, interpolated, and assigned to each farm. 4.4 Satellite Data explains the source of satellite data as well as how the dataset is built. 4.5 Masking shows how the combined cadastral and field data is used to generate accurate image masks for cultivated fields.

Chapter 5 presents the methodology of the thesis, including a general system architecture and details behind the implemented models to carry out the yield prediction. 5.1 Data Preprocessing describes the preprocessing steps of both the input and ground-truth values. 5.2 Weather Data presents a neural network designed to predict based on weather features. 5.3 Satellite Images describes two models using the satellite images to make predictions, and 5.4 Satellite Images and Weather proposes three models using a combination of satellite images and weather features. 5.5 Reducing Overfitting shows the techniques used to augment the dataset and combat overfitting.

Chapter 6 showcases the experiments conducted on satellite images, and comparisons of the implemented models' performances.

Chapter 7 concludes and summarizes our results and assess them against our problem statement and the four hypotheses.

Chapter 8 discusses additional directions for future work.

Chapter 2

Theoretical Background

The theory behind predicting crop yields based on satellite data spans multiple domains, and this chapter lays the foundation for topics and work covered later in the thesis. Section 2.1 Artificial Neural Networks explains the building blocks of neural networks and introduces the architectures utilized in the proposed models. Section 2.2 Plant Growth Factors briefly describes which elements that are important for sustained plant growth. Finally, Section 2.3 Remote Sensing explains what remote sensing is, how the data is typically collected, and why it is useful for analysing vegetation.

2.1 Artificial Neural Networks

2.1.1 The Perceptron

In 1958, Frank Rosenblatt published the paper "The perceptron: A probabilistic model for information storage and organization in the brain", in which the perceptron was invented [29]. Rosenblatt called the perceptron a "hypothetical nervous system", indicating that the perceptron was designed to represent some properties of intelligent biological systems.

The theory behind the perceptron was built upon of the following theories of Hebb, Hayek, Uttley and Ashby [29]:

- The physical connections in the nervous system involved in learning are not identical from one organism to another and is at birth constructed largely at random.
- A system of connected cells are capable of plasticity, which means that applied stimulus towards one set of cells will likely cause the response in other cells to change, due to long-lasting changes in the neurons.
- With exposure to large samples of stimuli, the stimuli which are the most similar will tend to cause stronger pathways to the same set of responding cells.
- Positive and negative reinforcement can enable or hinder formations of pathways between cells.

The perceptron architecture is based on the artificial neuron *threshold logic unit* (TLU) proposed by Warren McCulloch and Walter Pitts in 1943, and can be seen in Figure 2.1 [29][9]. The TLU accepts numbers as inputs, and each input is assigned a weight. After applying the weights to the input values, the weighted sum is calculated, and a step

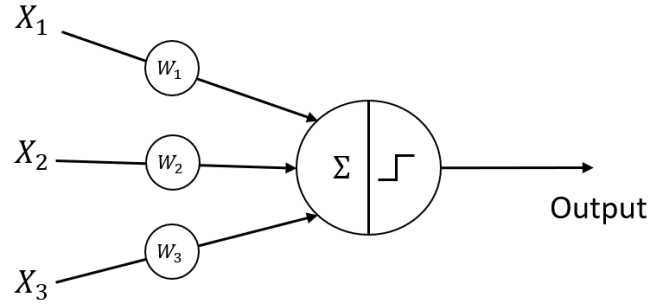


Figure 2.1: The threshold logic unit

function is applied, which gives the output. The step function could be a simple Heaviside step function, shown in Equation 2.1. The perceptron is trained by adjusting the weights assigned to each input. Since the perceptron contains a single computational layer, it can be considered a single-layer network [1].

$$\text{Heaviside}(x) = \begin{cases} 0 & \text{if } x < 0 \\ 1 & \text{if } x \geq 0 \end{cases} \quad (2.1)$$

Equation 2.1 illustrates the the heaviside step function.

2.1.2 Deep Neural Networks

Expanding on the idea that a single perceptron is a single-layer network, a *Multilayer Perceptron* (MLP) is constructed by combining multiple perceptrons and placing them in layers [1, p. 17], see Figure 2.2. An MLP has three main components: An input layer, one or multiple hidden layers consisting of TLUs, and the final layer of TLU making the output layer. The architecture seen in Figure 2.2 can also be called a feed-forward network, as the data flows from the inputs into each layer sequentially to finally reach the output.

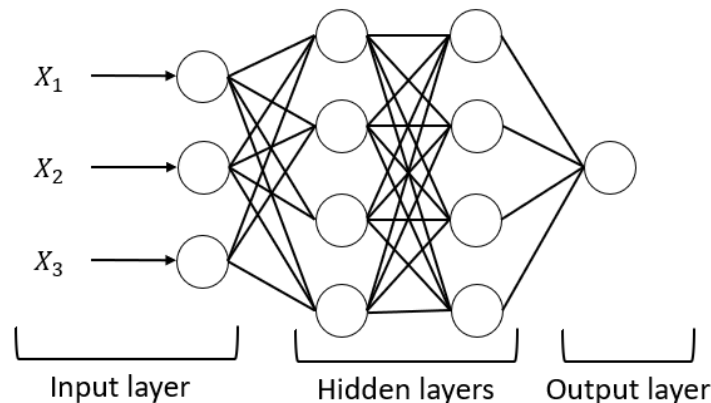


Figure 2.2: A Multilayer Perceptron

An MLP architecture can be considered a *deep neural network* (DNN) when the stack

of hidden layers is big enough, although the exact number of layers required for it to be considered *deep* is not clearly defined. [9]

2.1.3 Convolutional Neural Networks

In deep neural networks each neuron is directly connected to all the neurons in the previous layer, which allows each neuron to learn global patterns across its input space. However, while this approach allows fully connected layers to learn complex patterns on the input space as a whole, it also limits their ability to detect local correlations that can appear at any position in the input space, and it requires that the input must be presented in a fixed order. Theoretically, a fully connected network could learn to identify a local pattern in any position of the input space by simply adding enough neurons. However, this would likely lead to multiple neurons sharing the same weight patterns but located at various locations in the network, so they could detect the pattern at different locations in the input space [20]. This leads to an inefficient network architecture, which increases computational cost and requires large datasets that includes samples of the pattern in all possible locations.

In 1981, inspired by research on the receptive fields in the visual cortex of cats and monkeys, Kunihiko Fukushima created a new layered hierarchical architecture which he called the *neocognitron*, in which each neuron is only connected to the neurons of a small patch in the previous layer [7]. By gradually decreasing the size of the spatial dimension in the deeper layers, each layer increases the receptive field of its neurons. Neurons in the final layer of the neocognitron thus have a receptive field which indirectly cover the whole input space, and can respond to a specific pattern irrespective of it's position or size [7]. The neocognitron used a combination of convolution layers to extract features and subsampling layers to down-sample the input, and it was this combination of layers that later paved the way for convolutional neural networks [20].

Today's convolutional neural networks (CNNs) share the same basic architecture of the neocognitron, using a combination of convolutional layers and subsampling layers called pooling layers. They are widely used in the field of computer vision, taking images represented as 3 dimensional matrices of pixels (width \times height \times channels) as input. Both convolutional layers and pooling layers takes a 3D matrix as input and outputs a new 3D matrix, typically with fewer pixels than the previous layer.

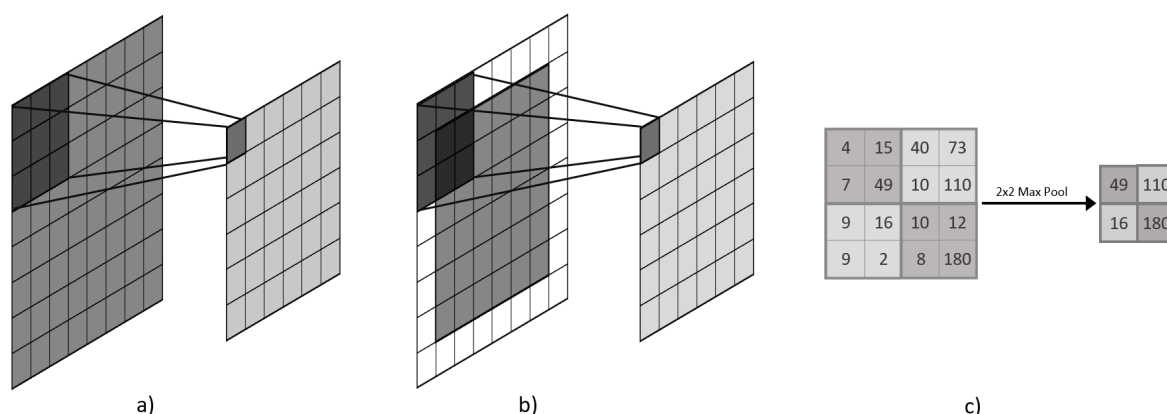


Figure 2.3: Illustration of convolutional layers. a) 3×3 convolutional filter. b) 3×3 convolution with padding. c) 2×2 Max Pooling.

A convolutional layer consists of a number of *filters*, each with its own set of trainable weights (a neuron) of size $m \times n$, that are applied to the whole input in a sliding fashion (see Figure 2.3a). A typical filter size of 3×3 means that each pixel in the output is created by looking at a 3×3 grid of pixels directly above it in the previous layer. By reusing the same weights, the filters *kernel*, in a sliding window across the input, a filter produces a feature map of a particular feature or pattern that it has specialized in [9]. For example, a kernel that activates strongly on edges will output a feature map that highlights pixels corresponding to an edge in the input, while other pixels are blurred out. When applying a convolutional filter, it is common to slide the kernel one pixel on the input at a time, or with a *stride* of one pixel. To keep the dimensions of the input unchanged, allowing deep architectures, convolutional layers often use a technique called 'same' padding, which pads the input with enough zero-valued pixels to maintain the same size as the input [39, p. 13] (illustrated in Figure 2.3b).

Some CNNs use strided convolutions, meaning that instead of moving the filter one pixel at a time, overlapping with the previous unit, we move with larger steps. Striding is useful for reducing the output dimensions of the layer, however, a more common way of reducing the output size is by using *pooling layers* [39, p. 96]. Unlike a convolutional layer, a pooling layer have no trainable weights, and is typically performed with a stride equal to it's size, resulting in a drastic reduction of parameters. The most common pooling, max pooling, is done by applying a max filter on each region it passes over, only returning the highest value of that region. A 2×2 max pooling layer with a stride of 2 will reduce the number of features by a factor of 4, as illustrated in Figure 2.3c, while keeping the strongest activations from the previous layer.

2.1.4 Recurrent Neural Networks

While traditional neural networks and convolutional networks are specialized for flat data structures and matrix data, recurrent neural networks (RNNs) are a family of neural networks that take data sequences as input. What separates RNNs from other networks is their ability to "*remember* an encoded representation of its past" [35] by passing on the output of the previous timestep along with the input at the current timestep. Each timestep of the sequence is processed using the same weights as all other timesteps, significantly reducing the number of neurons required to process long sequences of data. Reusing the same weights for each timestep also allows RNNs to generalize well even on sequences of varying lengths, as the output can be extracted from any step of the calculation.

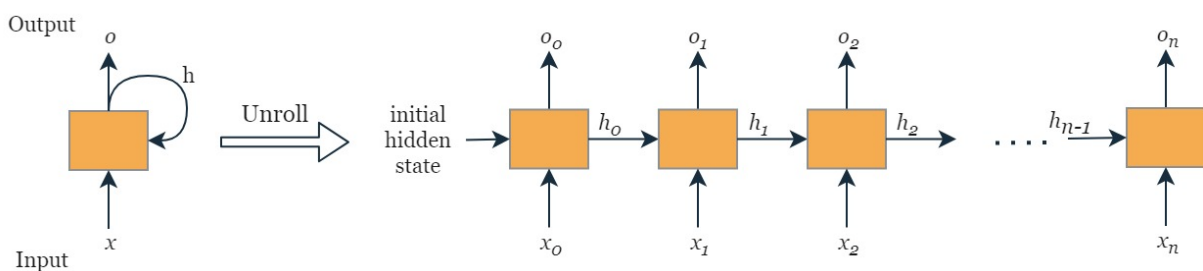


Figure 2.4: A Recurrent Neural Network illustrated as both a cyclic graph and unrolled.

As RNNs pass on the output of the previous timestep when processing the current timestep, the flow forms a directed graph, looping in on itself; for each step, it feeds a *hidden* state back into itself. When processing data one timestep at a time, it is easier

to visualize the graph unrolled into each separate timestep as illustrated in Figure 2.4. Each step produces an output vector (o) and a hidden state vector (h) that is passed on. As the hidden state is continually changed by the new information input at each step, it can be view as the encoded representation of all prior steps (i.e. the past). The output from the last step is similarly an encoded representation of the complete sequence.

In the most simple case, an RNN cell consists of only a single, fully connected NN layer. An illustration of a simple RNN cell is provided in Figure 2.5, with the common activation function \tanh . The previous hidden state is concatenated with the current input before being fed into the NN layer, which produces a new hidden state for the current timestep. The output of such an RNN is simply the hidden state at each timestep, or the last timestep, depending on the usage.

Long Short-term Memory

A known limitation of simple RNNs is their inability to retain information across long sequences of data because of the vanishing-gradient problem that arises with very deep neural networks [15][13]. To remedy these issues, Hochreiter and Schmidhuber released a significantly more complex RNN cell called the Long Short-term Memory cell (LSTM) (see LSTM cell in Figure 2.5) in 1997 [14]. In addition to the hidden state from each timestep, a cell state vector (c) is included to form a "gradient superhighway" that allows more information from the past to pass on to the next step [35]. The fully connected layer is also accompanied by three *gates* that allow the cell to better control the flow of information by learning which parts of the state and input are relevant at each step [14]. All three gates consist of, at minimum, a fully connected layer with Sigmoid activation and a pointwise multiplication that can open or close access for any element of the vector passing through. The forget gate determines, based on the previous hidden state and the current input, which parts of the state are relevant and which parts can be forgotten. The input gate controls which parts of the current input should be added to the cell state, and an output gate allows only currently relevant information from the cell state to be passed on as the cell output.

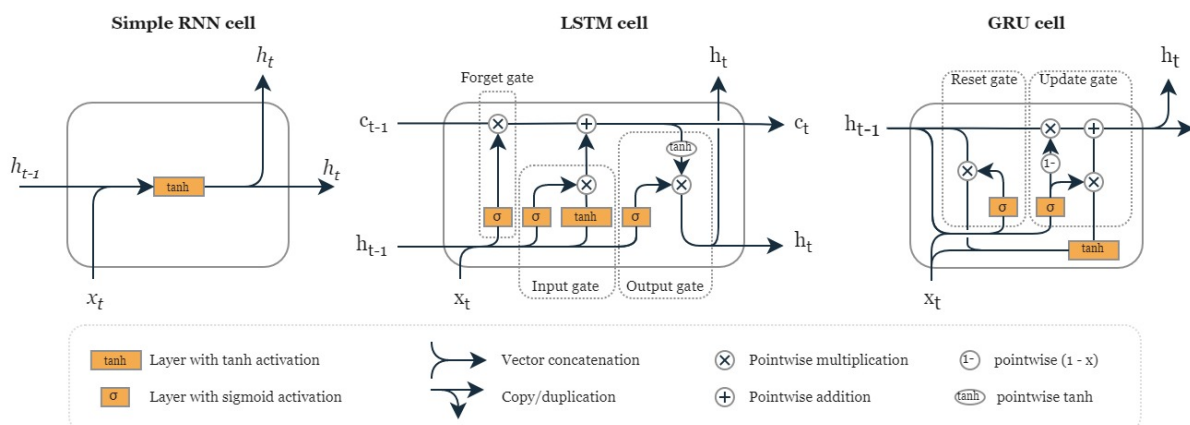


Figure 2.5: Illustrations of a simple RNN cell, an LSTM cell, and a GRU cell.

Gated Recurrent Unit

Although LSTMs have proven superior to standard RNNs, they are also more complex and require more computations to train. A newer variant of LSTM called the Gated Recurrent Unit (GRU) was designed to be simpler to compute and implement [6]. Compared with an LSTM cell, the GRU has no separate cell state and has only two gates: a reset gate and an update gate as shown in Figure 2.5. Both gates function similarly to the multiplicative gates of the LSTM. A reset gate controls which parts of the previous state should be ignored, and the update gate decides which parts of the hidden state should be updated with a new hidden state. These allow the GRU to drop information that is no longer relevant and to control how much information is carried over to the next step, which helps the RNN to remember long-term information [6].

2.2 Plant Growth Factors

A plant's growth and wellness are affected by elements from its surroundings. According to Oregon State University, the four main environmental factors affecting plants growth are: light, temperature, water, and nutrition [37].

2.2.1 Light

Light is a component of photosynthesis and is essential for overall plant growth. In Norwegian crops, the duration of light is particularly relevant; according to Åssveen and Abrahamsen, the duration of light in a day (day length) is more influential than temperature as growth factors [2].

2.2.2 Temperature

Temperature affects growth in several ways. The germination process is triggered by a rise in temperature, which means that the temperature controls when the seedlings initially sprout. The temperature also affects when crops such as winter wheat break dormancy to resume the growth in spring. A common measurement using temperature to estimate plant growth is the so-called sum degrees, meaning the sum of mean daily temperatures for the period. [37]

Crops will have different requirements for how much sum degrees are needed before it is ripe or ready for harvest, see Table 2.1 [2].

| Growth | Sum day degrees |
|---------------------------------------|-----------------|
| Barley, maturation | 1200 |
| Wheat and oat, maturation | 1600 |
| Grass for feed (before first harvest) | 750-800 |

Table 2.1: Sum degrees (in Celsius) requirements before harvest is ripe or ready for harvest [2].

2.2.3 Water

Together with light, water is a primary component of photosynthesis, and consequently, an essential factor for growth. For crops, water can come in the form of direct precipitation, humidity, or irrigation. [37]

2.2.4 Nutrition

Plants need in total 17 basic chemical elements to grow. Three of the required components are found in air and water (carbon, hydrogen, and oxygen), while the soil must provide the rest. Farmers can fertilize the soil, which adds materials containing nutrients so that these are available to the plants. The roots absorb approximately 98 percent of the nutrients through soil water. If the plant is under stress by extreme temperatures, drought, or low light, this can lower the plants' ability to absorb nutrients efficiently. [37]

2.3 Remote Sensing

Remote sensing is defined as "the field of study associated with extracting information about an object without coming into physical contact with it." [32]. Although the definition is rather vague and broad, the term is mostly used in the context of earth observations using optical imaging instruments on board satellites or aircrafts [32][33].

Satellites used for earth observations often carry a special optical imaging sensor, making them capable of measuring the earth's reflectance in multiple spectrums, far exceeding the visible spectrum. For example, the Sentinel-2 satellites carry a Multi-Spectral Instrument (MSI) measuring the reflectance of the earth in 13 spectral bands, from VNIR (visible and near-infrared) to SWIR (Short-wavelength infrared) [24]. The images produced by such instruments allow us to look at new parts of the earth and vegetation that is simply invisible to human eyes, as well as capturing changes over time that can span anything from minutes to decades of measurements [32].

Most remote sensing uses passive sensors to measure the surface, meaning that the sensor measures solar radiation that is reflected off the surface. As different materials and objects reflect light in different wavelengths, a multispectral image can be used to detect or measure a multitude of different features on the surface. By looking at the spectral reflectance curve of different surfaces (see Figure 2.6), it becomes apparent that the near-infrared and short-wave infrared spectrum contain much more information than is available in the visible spectrum alone.

Of particular interest is the spectral composition of vegetation, as well as the changes that occur over time, as plants grow and develop. For this purpose, special combinations of spectral bands have traditionally been used to create what is called vegetation indices.

2.3.1 Vegetation index

A vegetation index is a type of feature engineering in which several spectral bands are combined to form compact and more manageable vegetation features. A widely used type of vegetation index is the normalized difference vegetation index (NDVI). The NDVI uses known properties of vegetation and their reflectance to indicate whether or not a pixel from a multispectral image contains healthy vegetation and to which degree. The specific relevant properties used to calculate the NDVI are [27, p. 30]:

1. The leaves of plants contain chlorophyll pigments, which is an essential factor in photosynthesis and is ultimately what makes the leaves green. Chlorophyll pigments makes the leaves absorb a lot of the red and blue regions of the VNIR

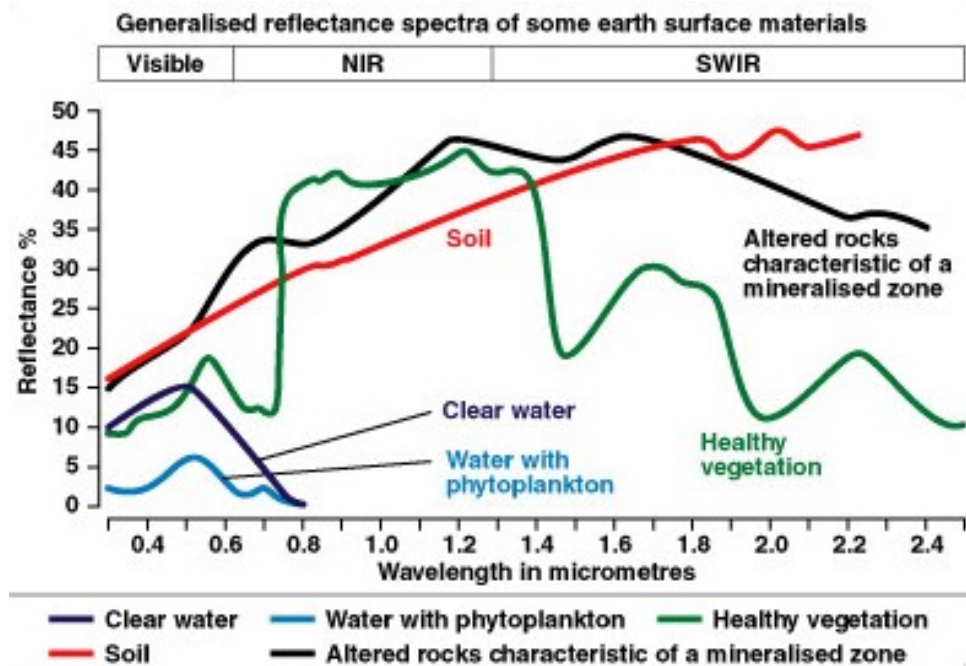


Figure 2.6: The spectral reflectance curves of different earth surfaces. Source: [28]

spectrum, but not in the green region. The number of chlorophyll pigments can indicate health in vegetation; thus, measuring the amount of reflection in the red spectrum can be used to estimate vegetation health. Low reflectance in the red spectrum indicates healthy vegetation.

2. Leaves have evolved to scatter solar radiation in the near-infrared (NIR) part of the spectrum, as it is difficult to extract the energy at these wavelengths efficiently (longer than 700 nm). This implies that healthy vegetation will have higher reflectance of NIR.

Based on these known properties, the formulae to calculate the NDVI for any given pixel can be seen in Equation 2.2, and Figure 2.7 shows NDVI applied to a satellite image. One method to utilize the NDVI values for existing models and frameworks is to get the NDVI of all relevant pixels and do an arithmetic mean to get a single value representing the NDVI of the entire area as a whole [4][17].

$$NDVI = \frac{NIR - R}{NIR + R} \quad (2.2)$$

Equation 2.2 illustrates how to calculate for the Normalized Difference Vegetation Index (NDVI) for any given pixel in a multispectral image. The pixel values from the near-infrared (NIR) and red (R) bands are used to calculate the NDVI for that specific pixel location.

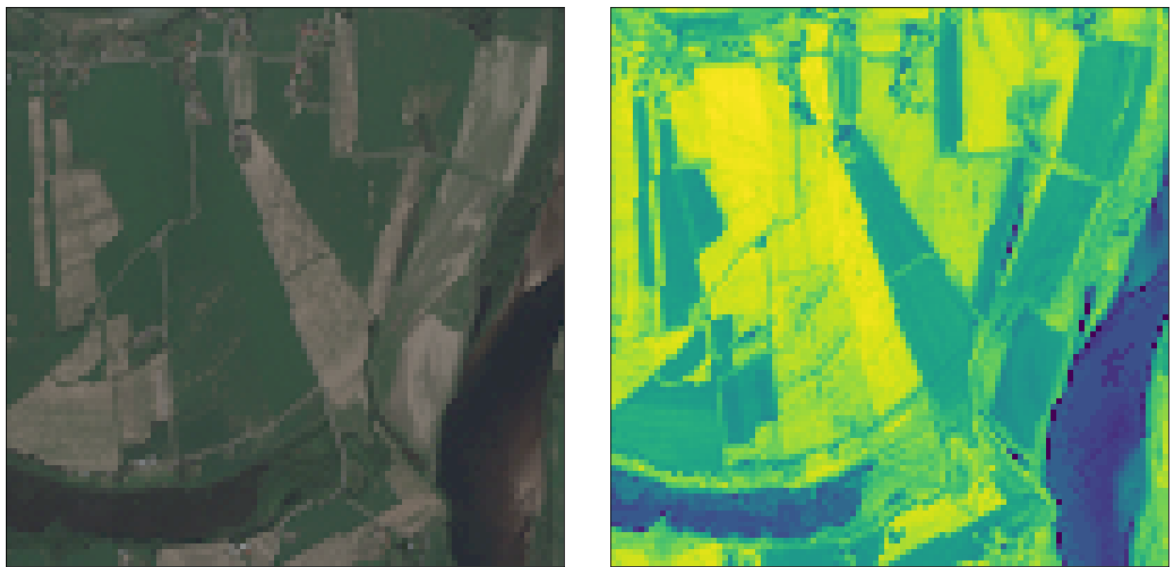


Figure 2.7: NDVI applied to satellite image of a farm. The image to the left is a representation of what can be seen using RGB, and on the right is a calculated NDVI representation of the same image. Water and empty fields have low NDVI and appear blue while healthy vegetation has higher NDVI values and appear yellow.

Chapter 3

State-of-the-art

Predicting crop yields is a well-established research area. Most research uses traditional statistical methods and handcrafted features derived manually from satellite images. In recent years, automatic feature extraction from multispectral images using deep neural networks have outperformed traditional methods that rely on handcrafted features. As such, deep learning has emerged as the current state-of-the-art for crop yield prediction using remotely sensed data [44][35].

This chapter investigates how the application and research using remote spectral observations for vegetation analysis have progressed over time. Section 3.1 Origins and Early Use of Remote Spectral Observations investigates the initial purpose of earth-observing satellites and the goals at that time. Section 3.2 Deriving value from Vegetation indices shows the era of handcrafted features derived from multispectral images and how these were utilized to make crop predictions. Section 3.3 Machine Learning Applied to Remotely Sensed Data explores how modern deep learning techniques are applied to multispectral images using both handcrafted features and automatically extracted features from raw data.

3.1 Origins and Early Use of Remote Spectral Observations

In 1973, NASA launched the world's first Earth-observing satellite named Landsat 1. The intent of Landsat 1 was to monitor and study the landmass of planet Earth; the satellite had two instruments to carry out the data collection: A primary camera system named the Return Beam Vidicon (RBV), and a secondary and experimental Multispectral Scanner (MSS) [19]. However, after studying the collected data, the roles of these two instruments changed, and MSS became the primary data source [19].

NASA oversaw 300 research investigators working on and exploring the data collected by Landsat 1 [19]. One significant project conducted by Rouse et al. in 1974 studied how the Landsat 1 MSS data could provide quantitative regional vegetative information of the farmlands throughout the Great Plains Corridor rangeland in America [30]. The study had three hypothesis[30]:

1. The vernal advancement and retrogradation of vegetation (green wave effect) can be discriminated on a regional basis using repetitive multispectral data.
2. Natural vegetation parameters provide a new information source for regional agri-business use.

3. Temporal effects are important in discriminating broad landforms, soil associations, vegetation types, and other natural resource features.

In the conclusion of this study, the researchers confirmed that the above hypotheses were correct [30].

3.1.1 A Large Area Crop Inventory Experiment

In 1975, America launched the Large Area Crop Inventory Experiment (LACIE) in a joint effort between the United States Department of Agriculture (USDA), the National Oceanic and Atmospheric Administration (NOAA) of the Department of Commerce, and the National Aeronautics and Space Administration (NASA). The project's goal was to evaluate and prove the economic importance of applications built using remote sensing from space. The LACIE project would concentrate on wheat grown in North America and combine Landsat data with meteorological information from NOAA to run experimental investigations on the crops. The ultimate objective would be to satisfy the requirements of being able to monitor and make crop production inventories on a global scale. [22]

In order to select which areas to study, the experiment used random sampling employing 5 x 6 nautical mile segments randomly allocated in areas known to plant wheat. In total, 1720 sample segments were selected, each segment containing 117 lines of 196 pixels. The satellites passed over each segment every 18 days. Due to cloud cover, the likelihood of the segment being visible was only approximately 60 percent. The weather data was collected from stations of the World Meteorological Organization (WMO), and a thirty-day average was used for the yield models [22][23].

The accuracy goal of LACIE was a 90/90 at-harvest criterion for wheat production. This meant that the production and area estimate that was made at harvest for a region or county should be within 90 percent of the truth, 90 percent of the time, i.e., 9 years out of 10. [23]

The results of LACIE were compared against the Statistical Reporting Service (SRS) to determine the accuracy of the estimates. The resulting wheat area estimates were deemed marginally satisfactory with regards to the 90/90 at-harvest goal. Furthermore, combining the area estimates with the yield estimates, the resulting production estimates met the 90/90 goal with about a 5.6 percent deviation from the actual production. [23]

3.1.2 Using Remote Sensed Information as Input for Agrometeorological Models

In 1976, the Agricultural Research Service (ARS) of the USDA continued developing agrometeorological models for forecasting wheat yields [43]. Even though the possible use of remote spectral observations for forecasting models received skepticism at the time, earlier research by Rouse et al., C. L. Wiegand, the LACIE project demonstrated that it would be both of value and technically feasible [30][43][23].

The spectral observations were to be collected through handheld, aircraft-, or spacecraft-mounted sensors with a goal of combining these observations with soil property and daily increments of weather data to ultimately estimate the yield of the saleable plant parts. Experiments conducted demonstrated that the spectral observations could be used to calculate vegetation indices, which could be used to measure

amount of green photosynthetically active tissues as well as estimate reliably the leaf area index (LAI), both of which could be used as input to their models [43].

In an evaluation of the progress made from the period 1976 to 1986, Wiegand found that spectral observations in conjunction with agrometeorological models increased confidence of models, however he also argued that the newness of spectral interpolations together with continual revisions in the agrometeorological models prevented the benefits of spectral inputs to be fully realized [43].

3.1.3 Summary

Earth-observing satellite data provided by NASA opened the possibilities of using multispectral imagery for analytics in agriculture. As proven by Rouse et al. [30], C. L. Wiegand [43], and the LACIE project [23], these images carry information that is useful for crop yield-related experiments, but it is challenging to extract the full potential of the data. The most promising avenue at the time was the use of vegetation indices.

3.2 Deriving Value from Vegetation Indices

After the initial research and experiments on vegetation and yields using MSS data collected by Landsat, there was further work to continue these efforts. One common and central theme seems to revolve around the use of vegetation indices [17] (see Section 2.3.1 for details). Vegetation indices can capture the information of multispectral images into a format well suited for experimentation using existing models and techniques.

3.2.1 NDVI to Estimate Wheat Yield in Italy 1986-1989

Bendetti et al. conducted a study to investigate the potential use of NDVI applied to spectral imagery collected by NOAA satellites in Italy in the period 1986 to 1989 [4]. The study considered the production of wheat of the provinces in the region Emilia Romagna. The spatial resolution of the images collected was about 1x1 km and was applied to test sites of 900 ha (3x3km), resulting in each test site having 3 pixels x 3 pixels of data per image. The researchers calculated the NDVI of each pixel, and the mean of these represented the real NDVI for the study site.

The data were grouped into 10-day intervals, and the maximum NDVI of every ten days was chosen to represent that period. The researchers applied a linear regression model to these data and observed a production estimate within 10 percent of the official registered production overall. In the conclusion of this study, the researchers states that it is safe to assume that this methodology provides good yield estimates at a province and regional scale [4].

3.2.2 Relationship of NDVI and Weather Features to Corn and Soybean Yields

In 2014, Johnson assessed the use of remotely sensed variables for forecasting corn and soybean yields in the United States. In this case, the remotely sensed variables were: Satellite multispectral images collected from Terra satellites, daytime and nighttime land surface temperature (LST), and precipitation. The satellite images were masked based on the Cropland Data Layer (CDL) produced by the NASS, such that

only pixels connected to soybean or corn remained, and an NDVI was calculated for each pixel. Much like Bendetti et al., the representative NDVI of each day was the mean of all NDVI values from an image [17].

To investigate the relationship between these data and yield, Johnson grouped them into 8-day intervals. He took the averages of NDVI, daytime LST, and nighttime LST together with the accumulated precipitation of this period, such that each interval had four features. One year had 32 of these intervals (from mid-February through late October), and the seasons considered were from 2006 to 2011. To explore these dependencies, Johnson used *The Pearson product-moment correlation coefficient*. Surprisingly, the results indicate that both precipitation and nighttime LST had little to no correlation to the yield. Johnson argues that irrigation or possibly build-up of moisture in the soil makes direct rainfall not as essential as first thought, and nighttime LST does not seem to affect plant growth as most active growth happens during daytime. NDVI was found to be strongly and positively correlated, while daytime LST was negatively correlated [17].

Johnson used 32 features for daytime LST and NDVI together with variables for the state, county, and year to estimate yield. Using Rulequest Cubist¹ he saw a root mean squared error (RMSE) between 1.26 and 0.96 metric tons per hectare of corn yield estimates.

3.2.3 Summary

The use of vegetation indices has been proven to work adequately with crop yield prediction and estimation. Given that satellites had a relatively poor resolution for many years, using vegetation indices such as NDVI made it possible to capture vegetational relevant properties into a single feature, well suited for various linear models.

3.3 Machine Learning Applied to Remotely Sensed Data

There has been increasing use of machine learning and deep learning techniques on remotely sensed data to estimate and predict different crop yields, and it appears to hold great potential [5][44]. Various studies have found that, generally, non-linear approaches outperform linear models when predicting and estimating yield with remotely sensed data [5][16][17].

3.3.1 Growth Phases as Timesteps in LSTM

Crop phenology² varies from season to season depending on environmental and managerial factors. Jiang et al. set out to explore a phenology-based LSTM model for corn yield estimation [16]. The corn crop includes six distinct phases of development throughout the season: planted, emerged, silking, dough, dent, and mature. Jiang et al. split these into five growth phases (GP)³, where one growth phase symbolizes one timestep for the LSTM.

¹Rulequest Cubist is a tool to data-mine and extract patterns from data, trying various linear models and forms an ensemble of the best fitting models, which can be used for predictions.

²Phenology is defined as "a study of the timing of recurring biological events" [41].

³The growth phases as identified by Jiang et al. GP1: planted to emerged; GP2: emerged to silking; GP3: silking to dough; GP4: dough to dented; GP5: dented to mature.

Each timestep included three meteorology features and a single vegetation index WDRI⁴ (in total 4 features x 5 timesteps), with this the LSTM should estimate county-level corn yield. With ten years of training data (2006-2015), they saw a RMSE of 0.87 metric tons per hectare [16]. This result is better than what Johnson [17] could manage using the rulequest cubist (0.96) but cannot be directly compared due to the number of and which seasons involved in the training is not the same.

3.3.2 Satellite Image Pixel Histograms as Input to CNN and LSTM

In 2017, You et al. published their novel convolutional and LSTM network for predicting soybean yield on a county-level scale in the USA using multispectral images [44]. As far as we know, their research is the first to use the raw images as input to the deep learning algorithms. You et al. argues that using handcrafted features such as NDVI can be fairly crude, and that using deep learning to find the relevant features in multispectral images automatically can be more effective.

Their approach assumes a *permutation invariance*, meaning that the position of a pixel is considered to be less relevant to the average yield in a patch. From this assumption, it is argued that a multispectral image can be compressed into histograms of pixel values without loss of information. They collected the surface reflectance, land surface temperature, and land cover type for locations in the US with soybean production, resulting in 9 channels of multispectral data. They collected the data from the 49th to the 281st day of the year, with 8-day intervals, making 30 images per year. The images were converted to histograms of pixels with 32 bins, resulting in 32 (bins) x 9 (channels) x 30 (images) features.

Both convolutional and LSTM networks were trained on these features. They saw that these networks significantly outperformed competing methods, such as ridge regression, decision trees, and DNNs, with an RMSE reduction of 30 percent compared to the best of the competing models. You et al. demonstrates that deep learning models can automatically find relevant features for yield prediction from multispectral imagery and that handcrafted features might not be necessary. [44]

3.3.3 Using Raw Satellite Images to Predict Wheat Yield in India

Inspired by the result of You et al. and their histogram approach, Sharma et al. trained neural networks using raw satellite images into a CNN-LSTM model. Their approach forgoes any handcrafted or rudimentary features such as vegetation indices or histograms. Instead, they use a convolutional neural network to perform all necessary feature extraction and learn the best representation that is useful for yield prediction. They argue that prior work has not taken into account surrounding factors such as water bodies or urban areas that may affect crop yield. [35]

The model of Sharma et al. takes 300x300 size images, with nine spectral bands including VNIR, SWIR, and thermal bands, all with a 500m resolution per pixel. In addition, they provided three land cover masks as three separate channels: water bodies, agriculture, and urban areas. The masks were created by classifying on a pixel-by-pixel basis, each pixel assigned to a class if 60% or more of the pixels area correspond to that class. The images are fed through a five layer CNN, before being fed

⁴Wide dynamic range vegetation index (WDRVI) is a vegetation index similar to NDVI, however, will be less affected by the saturation effect when density of biomass is high [16].

into a three layer LSTM network and three fully connected layers to perform the yield prediction.

Training samples were created from wheat yield in tehsil level blocks in the northern parts of India. A tehsil is a small administrative unit in India, and the average size of the included tehsils were 35,000 hectares. By comparing their model with the histogram approach of You et al, as well as other machine learning and regression methods, they show that using raw satellite images outperforms all previous methods on these data. In addition, they show that the addition of land cover masks improved their model by 17%, which suggest that contextual information is important for these types of models.

3.3.4 Field-scale Yield Prediction of Corn and Soy

All mentioned studies in this chapter predict and estimate the yield on a county or regional scale. In March 2021, Sagan et al. published a study specifically investigating the use of raw satellite images for field-scale level yield prediction. Although You et al. used raw satellite image data, they also condensing the data using pixel value histograms. As well as to investigate the use of raw images, Sagan et al. also did experiments based on several handcrafted features such as vegetation indices. [31]

The data for such fine-scale yield data were collected at the University of Missouri Bradford Research Center (BRC) in 2017. They used three experimental fields, two for corn and one for soybean, and split them further into 293 plots of corn crops and 216 plots for soybean crops. The size of each plot varied slightly from crop to crop and was approximately 0.76 m x 8.2 m.

To gather remotely sensed data, they used two satellites (WorldView-3 and PlanetScope) for multispectral images and a drone (Mavic Pro quadcopter) for high-res RGB images. The satellite images were re-sampled to 0.3 m per pixel resolution, and the individual plots consisted of 28 x 28 pixel multispectral images.

Sagan et al. used the images collected from each plot for two main directions in their study: 1) Condense them into handcrafted vegetation indices, and 2) Use the raw images directly in a CNN-based model. Their results show that raw image-based deep learning performance was comparable, if not superior, to deep learning methods using handcrafted features. Overall the percent root mean square error was about 10 percent regardless of crop-type and irrigation conditions. Their work showcase that an image-based deep learning approach can utilize spectral, spatial, and temporal information from the satellite data, and essentially reduce the need for feature engineering.

3.3.5 Summary

There have been two significant developments within the use of remotely sensed data for yield prediction in the last five years. Firstly, non-linear models such as deep neural networks seem to outperform the linear regression methods typically used in the past. Secondly, researchers have started to re-examine the aspect of extracting information from multispectral images. You et al. paved the way by using data from satellite images without the use of handcrafted features, but instead with images compressed into pixel histograms. These histograms were used as input to train deep learning models, and their results show that these models performed yield prediction better than earlier methods, indicating that the models were able to find important features

automatically. Sharma et al. continued along these lines by directly using the raw images as input to their models and reported even further improvements for crop yield predictions.

Thus far, most studies have evaluated the use of remotely sensed data on a county or regional scale, as crop yield statistics required to make predictions based on farm or field-scale have not commonly been available to the general public [34]. Sagan et al. made an effort to predict field-scale crop yields using deep learning by building a dataset consisting of small experimental plots and making yield predictions for these plots. Their results show that the models can learn growth-related features even on such small plots, indicating that remote sensing with deep learning can be effective for both field and farm-scale predictions, given enough crop yield statistics are available.

Chapter 4

Multi-Temporal Data and Data Handling

There is no easily available single dataset that can be downloaded and used for machine learning in the field of yield prediction in Norwegian agriculture. Therefore, a large portion of this work has been to collect and connect data from different sources that can be used for crop yield prediction on a farm-scale level. This chapter explains which data sources are used and how the data has been collected.

4.1 Norwegian Agriculture Agency

The main data sources for the project are the official public archives of farmer grant applications and grain deliveries from the Norwegian Agriculture Agency, from which yield prediction targets can be made. As Norwegian grain farmers rely on subsidies, they are required to fill out yearly grant applications describing the land used for crop cultivation. Some of these data are publicly available, and these are used to build a base dataset upon which other data sources can be connected through each farmer's unique organization number¹.

From the Norwegian Agriculture Agency, there are three different yearly reports that serve as the base data from which we build our dataset, these are the grain delivery reports, agriculture production subsidies, and a land use report that connects each farm to one or more cadastral units².

4.1.1 Grain Deliveries

The yearly grain delivery reports include how much grain of different types each farmer has sold in the last year. The deliveries are categorized into 18 different distinct categories, which separates grains sold as food, animal feed, and seeds. The reports include deliveries for five different grains, as well as oilseeds and peas. As this work will focus on grain yields, only numbers for the grain crops are used. The grain crops are:

¹All farmers are registered in the official registers (*Brønnøysundregistrene*), giving them a unique organization number.

²An area of land, as specified in the official Norwegian cadastre (*matrikkelen*).

- Barley
- Oat
- Wheat
- Rye, and rye wheat

All the deliveries are stated in kg and are categorized as either animal feed, seeds, or for human consumption based on the quality of the grains. To create prediction targets, all three categories are summed into a single yield value for each grain type. Deliveries for rye and rye wheat are reported separately but are summed into a single category when used to calculate target yield, because farmers only report the combined cultivated area for these grains.

4.1.2 Production Subsidies

The second crucial yearly report contains agriculture production subsidies applications from farmers, which contain more than a hundred different reported data points for each farm. The relevant data used in our experiments are the area of cultivated land per crop type, stated in decares (daa). Together with the deliveries, these areas give the relative crop yield for each farm and crop type. Relative crop yield (kg/daa), as opposed to total yield (kg), have shown better results when used as the prediction target for simple neural networks, even when the area is provided as an input feature [36].

The published report format of the production subsidies underwent some changes from 2016 to 2017, in which more data points were added. Previous to 2017, all information about the area of cultivated land was not included in the published reports. Only years 2017, 2018, and 2019 have enough available data to create datasets for yield prediction as of May 2021. Area of cultivated land use is the only missing information that prevents data from 2012, through 2016, from being used in our experiments.

The production subsidies also provide a cadastral identifier for the main property used by each farm. This allows an easy connection to the cadastral data discussed later and could be used to extract the position of farms. However, as many farms span multiple cadastral units, the main property has only limited usefulness as it not sufficient for precise masking of images.

4.1.3 Land Use

As farmers are required to submit which land areas are used, the Norwegian Agriculture Agency has detailed reports linking farmers' organization numbers to all used cadastral units. These reports are not as readily available as the two previously mentioned reports and must be actively requested. Access to land use reports from 2017 to 2019 have been acquired³ and serves as a basis for precise geographic location and masking of images as described in sections 4.2 and 4.5 respectively.

³The land use reports were made available to us through the involvement in the Kornmo project.

4.2 Geographical Data

Precise geographical mapping for each farm is required to retrieve accurate remote sensing data. While weather data is possible to get with an approximate position from the sparsely located weather sensors, satellite images of a farm demand knowing the farm's exact geographical location and extent. The Norwegian cadastre comprises a publicly available geographical map layer of all properties, uniquely identified using a commune number and cadastral identifier, and the Norwegian Institute for Bioeconomy Research (NIBIO) provides a map of cultivated land areas throughout most of Norway. Together with cadastral identifiers from the land use data, these two map layers allow us to create precise geographical mappings for each farm in the dataset. Figure 4.1a and 4.1b show a visualization of the two map layers, further explained in the following subsections.

4.2.1 Cadastral Data

The Norwegian cadastre contains millions of geographical entities, each one describing some land area with a unique label or identifier. The cadastral identifiers are composed of four parts: a commune number and three numbers describing the exact property and section. Using the land-use reports from the Norwegian Agriculture Agency, a collection of geographical shapes for all farms, each year from 2017 to 2019, was extracted from the cadastre. The collection serves as a mapping from a farm's organization number and production year to its geographical extent. However, as the cadastre lacks information about agricultural land, the geographical data does not separate forested areas, water, and other land types from cultivated land (visible in Figure 4.1a).

Early experiments and about half of the satellite image dataset were downloaded by using the position of the main cadastral unit from the production subsidies reports (see Section 4.1.2). This approach, while not ideal, allowed experiments and preparations for later experiments using more accurate images when better land masking was available, and as the main property is often near the other land used by a farm, it was possible to reuse the images with the newer land masks.

As the last three numbers of the cadastral identifiers are unique only within each commune, the Norwegian communal reform meant that most identifiers in reports from 2017, 2018, and 2019 were outdated. New and updated cadastral identifiers for old values are obtainable through a public API at Geonorge⁴, which was used to create a mapping from the old to new identifiers compatible with the latest cadastre.

4.2.2 Field Boundaries

Norwegian Institute for Bioeconomy Research (NIBIO) keeps and maintains catalogs of geographical data associated with Norwegian agriculture. NIBIO tracks soil and field-related features such as quality, organic material, and water storage capacity throughout Norway. These datasets are made publicly available by Geonorge⁵ and can be downloaded in formats such as gdb (Geodatabase).

The datasets from NIBIO have geographical information about cultivated farmland in Norway, making it possible to use these datasets to distinguish fields from the rest of

⁴<https://ws.geonorge.no/kommunereform/v1/>

⁵<https://www.geonorge.no/>



Google maps base map, for comparison with images below



a) Cadastral layer, containing only registered grain farmers



b) Field boundaries from NIBIO soil quality dataset



c) Intersection of cadastral layer and field boundaries

Figure 4.1: Visualization of the geographical layers, overlaid on a satellite map.

the environment (such as lakes, forests, and towns). A well-suited dataset for this is the soil quality⁶ dataset, illustrated in Figure 4.1b, which according to NIBIO, maps roughly 50 percent of all cultivated fields in Norway [26].

The soil-quality dataset includes detailed information of the field boundaries and further classification of soil quality within these fields. For this project, only the field boundaries are used.

4.2.3 Combined Cadastral and Field Data

To connect field boundaries from NIBIO with the cadastral boundaries data from the cadastre, a new geospatial dataset is created by extracting the intersection between the two layers while keeping cadastral attributes. The output dataset has precise field boundaries for each farm. See Figure 4.1 for illustrations of the two base layers (cadastral and field boundaries) and the resulting intersection.

4.3 Weather Data

The weather is one of the main external factors that are crucial for farming. Grain farmers depend on periods with little precipitation in the spring so that the fields are dry enough to support heavy equipment for plowing, harrowing, and sowing. After sowing, the temperature must be stable so that the seedlings sprout, and precipitation throughout the summer is required to water the plants. As the grains mature, again, a period of limited precipitation is needed so that a combine harvester can harvest the grains before they can be delivered to the mills.

In addition to directly affecting the practicalities of farming, weather data includes two of the four main factors of plant growth [37]: precipitation and temperature. As shown by Johnson [17] temperature is highly correlated to the eventual yield, and precipitation is relevant when there is no irrigation used[18].

4.3.1 Collecting Weather Features

The Norwegian Meteorological Institute (MET Norway) collects weather data across Norway, and makes it publicly available through the Frost API⁷. In total, MET Norway has 1578 weather stations throughout Norway, where roughly 840 includes temperature and 630 includes precipitation data, with some variations from year to year.

The temperature measurements are available at one-hour intervals, but that level of granularity is not required for this project. The temperature measurements are split into individual days, where a min, max, and arithmetic mean are stored. This results in three temperature features per day. Precipitation data is the accumulated precipitation per day. The Frost API allows queries directly to get these readings, and the accumulated precipitation is downloaded from each of the weather stations with these measurements, resulting in one precipitation feature per day.

⁶In Norwegian: *jordsmonn*

⁷<https://frost.met.no/>

4.3.2 Interpolation

Previously, the weather at each farm was estimated using the readings from its nearest weather station [36]. This approach has been further improved by including weather stations with intermittent data. However, due to the limited number of weather stations throughout Norway, many farms had identical weather data as they were nearest to the same weather station. Additionally, farms who weren't close to any specific sensor were likely to have a larger difference in estimated versus actual value.

To reduce this difference, interpolation using neural networks is used to estimate the weather at each farm.⁸ By training two deep neural networks (DNN) to create soft sensors, lower deviations are achieved than with nearest neighbour interpolation. Training samples are created by keeping the reading of a sensor as the actual/desired output value, and providing the readings of the three closest sensors, as well as their normalized latitudinal, longitudinal, and vertical (distance from sea level) differences as inputs. The trained model is then used to create soft sensors at the location of each farm.

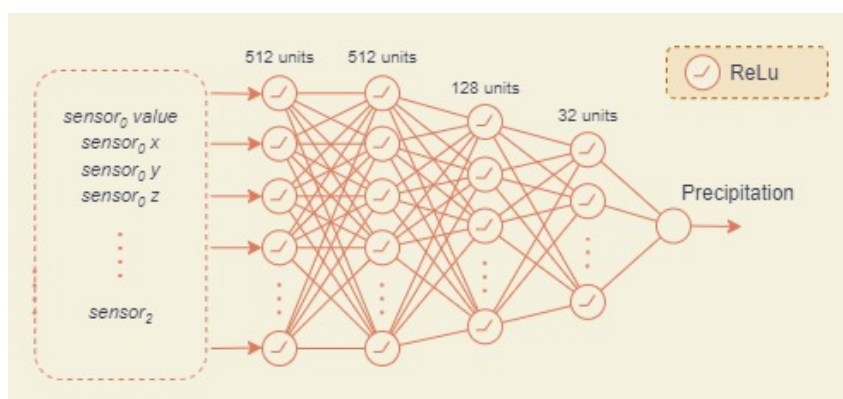


Figure 4.2: The neural network architecture used for precipitation interpolation

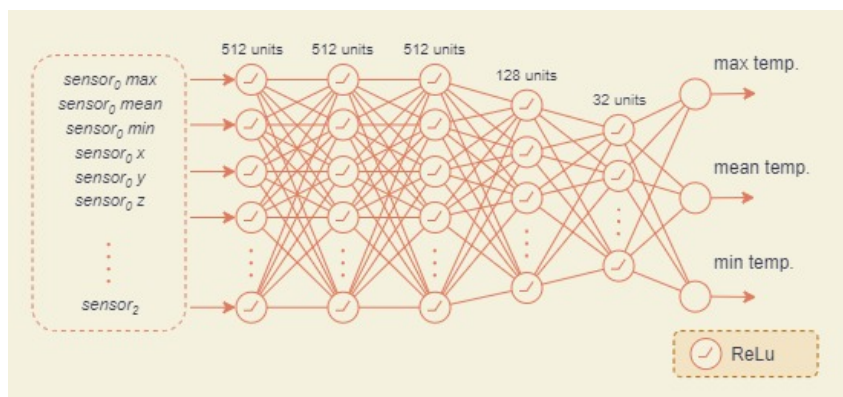


Figure 4.3: The neural network architecture used for temperature interpolation

Temperature and precipitation models are trained separately since the temperature and precipitation sensors often have different geographical locations. A deeper network showed slightly lower prediction error for the temperature model (Figure 4.3), as opposed to the precipitation model (Figure 4.2), where additional depth provided

⁸Interpolation using linear triangulation was attempted but was dismissed because too many farms are outside the bounds of the possible triangulation arrangements.

no significant benefit. As shown in Table 4.1, when compared to nearest neighbour, the DNN models achieve 23% reduction in mean absolute error in precipitation soft sensors, and 67% reduction in mean absolute error in temperature soft sensors.

| | Nearest Neighbour | Deep Neural Network | Change |
|----------------------------|--------------------------|----------------------------|-----------------|
| Precipitation error | 1.5 mm | 1.15 mm | -0.45 mm (-23%) |
| Temperature error | 1.6 °C | 0.52 °C | -1.08 °C (-67%) |

Table 4.1: Mean absolute errors in weather interpolation

Training the previous model⁹ with the interpolated weather data improved predictions significantly, shown in table 4.2.

| | Nearest Neighbour | Deep Neural Network | Change |
|--------------|--------------------------|----------------------------|-----------------------|
| Run 1 | 87.58 kg/daa | 83.21 kg/daa | |
| Run 2 | 89.71 kg/daa | 83.74 kg/daa | |
| Run 3 | 87.94 kg/daa | 81.54 kg/daa | |
| Run 4 | 90.41 kg/daa | 82.24 kg/daa | |
| Run 5 | 87.88 kg/daa | 84.48 kg/daa | |
| Mean | 88.70 kg/daa | 83.04 kg/daa | -5.66 kg/daa (-6.38%) |

Table 4.2: Mean average errors of predictions by reusing the preliminary project's model. The individual runs are not directly comparable as the training and validation data splits are different.

4.4 Satellite Data

The use of remote sensed data, or satellite images, is the current state-of-the-art for crop yield prediction without intrusive and labor-intensive monitoring. Therefore, it is also a major focus of this thesis. Building a dataset of multispectral satellite images for farm-scale crop yield predictions relies on the availability of high resolution satellite images combined with precise geographical information about farms.

4.4.1 Sentinel-2

The Copernicus Sentinel-2 satellite mission is the source of all images used in our experiments. The Sentinel-2 mission is developed by, and operated by, the European Space Agency, and is comprised of two polar-orbiting satellites (S2A and S2B) that provide high resolution images of the earth every 5 days at the equator, and more frequent at higher latitudes¹⁰. Images from these satellites were accessed through Sentinel Hub, a subscription-based cloud API for satellite imagery¹¹.

The Sentinel-2 satellites carry a multispectral instrument (MSI) that captures optical images in 13 spectral bands, including visible light (red, green, and blue channels), NIR, and SWIR. The mission provides two main products or product levels: Level 1C and Level 2A. Level 2A provides access to atmospherically corrected surface reflectance

⁹A dense neural network built for the preliminary project [36]. It is worth noting that the inclusion of weather stations with intermittent data improved predictions compared to the previous report.

¹⁰<https://sentinel.esa.int/web/sentinel/missions/sentinel-2>

¹¹<https://www.sentinel-hub.com/>

values for 12 of the spectral bands, which is meant for use by other applications without further processing, and is the product used to collect satellite images for this project. The 12 spectral bands of the Level 2A product offering, their respective sensor resolution, and a sample gray-scale image for each channel is shown in Table 4.3.

4.4.2 Building a Multi-Temporal Image Dataset

Using the geographical shape files described in section 4.2, each farm's shape is converted into a point at the geometry's centroid, which is then used to build a 2 km x 2 km bounding box describing the bounds of the images for downloading. With 2 km x 2 km bounding boxes, roughly 65 % of all farms in the dataset are >90 % covered. This size provides good coverage of farms in the dataset while not being too large, which would lead to larger image size or reduced resolution (in meters per pixel). See table 4.4 for comparison between different bounding box sizes and how many farms would be covered.

In preparation for the experiments, a dataset of 509 910 unique Sentinel-2 images was downloaded and stored. Each downloaded image is of an area approximately 2x2 square kilometers, centered on a single farm. The resolution of the images is 100x100 pixels, meaning a single pixel roughly represents an area of 20x20 square meters. Upscaling of the 10m resolution bands and downscaling of the 60m resolution bands are handled by the cloud API, using nearest neighbor interpolation, such that all 12 channels were of equal size.

Fully utilizing the available 10m resolution bands with higher resolution images could potentially benefit the experiments. However, due to the multispectral nature of these images, they contain four times as many values per pixel compared to normal three-channel (RGB) images, increasing the download duration, processing time, API cost, and storage requirements. With a 20m pixel resolution, six out of twelve channels are stored in their native resolution, four channels are upscaled from 10m resolution, while only two bands are downscaled from 60m to 20m resolution. The 20m resolution requires a minimal amount of image processing (affecting API cost) while keeping the download time and storage requirements within reasonable limits.

Temporal changes for each farm in the dataset is captured by having multiple images of the same farm throughout the growing season. A total of 30 images are downloaded per farm, with a mean temporal resolution of 7 days. The download period is from March 1st to October 1st each year, split into 30 7-day periods for which the best image is queried, based on least cloud coverage¹². The result is an image time series for each farm, with weekly images from approximately week 10 to 39 (see Figure 4.4).

4.5 Masking

Image masking makes it possible to focus on portions of an image that are of interest. Masks can be applied so that it either highlights certain parts or remove irrelevant parts from the image. In the context of this project, provided we know where the cultivated fields are located, masks can be used on the satellite images to remove everything not registered as a cultivated field or as extra information to highlight where the fields are.

¹²Selecting the least cloudy image for every 7-day period is handled by the Sentinel Hub API






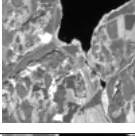
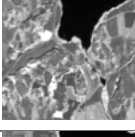
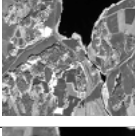
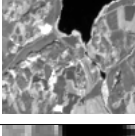

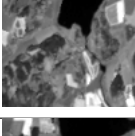
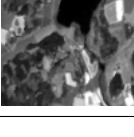
| Name | S2A/S2B Wavelength | Description | Resolution | Sample |
|-------------|---------------------------|---------------------|-------------------|---------------------------------------------------------------------------------------|
| B01 | 442.7/442.3 nm | Coastal aerosol | 60m |  |
| B02 | 492.4/492.1 nm | Blue | 10m |  |
| B03 | 559.8/559.0 nm | Green | 10m |  |
| B04 | 664.6/665.0 nm | Red | 10m |  |
| B05 | 704.1/703.8 nm | Vegetation red edge | 20m |  |
| B06 | 740.5/739.1 nm | Vegetation red edge | 20m |  |
| B07 | 782.8/779.7 nm | Vegetation red edge | 20m |  |
| B08 | 832.8/833.0 nm | NIR | 10m |  |
| B8A | 864.7/864.0 nm | Narrow NIR | 20m |  |
| B09 | 945.1/943.2 nm | Water vapour | 60m |  |
| B11 | 1613.7/1610.4 nm | SWIR | 20m |  |
| B12 | 2202.4/2185.7 nm | SWIR | 20m |  |

Table 4.3: The spectral bands of Sentinel-2 available from Level 2A, along with a single-channel greyscale image illustration created from each band.

| Bounding box size | Farms covered (>90 %) | |
|---------------------------|-----------------------|------------|
| | count | percentage |
| 0.5 · 0.5 km ² | 3 476 | 13 % |
| 1.0 · 1.0 km ² | 11 275 | 43 % |
| 1.5 · 1.5 km ² | 15 136 | 57 % |
| 2.0 · 2.0 km ² | 17 159 | 65 % |
| 2.5 · 2.5 km ² | 18 337 | 70 % |
| 3.0 · 3.0 km ² | 19 186 | 73 % |
| 3.5 · 3.5 km ² | 20 028 | 76 % |
| 4.0 · 4.0 km ² | 20 705 | 79 % |

Table 4.4: Different bounding box sizes and the number of farms that would be at least 90 % covered by such a bounding box. Coverage is calculated by the area of the intersection between the bounding box and the farm, divided by the area of the farms full extent.

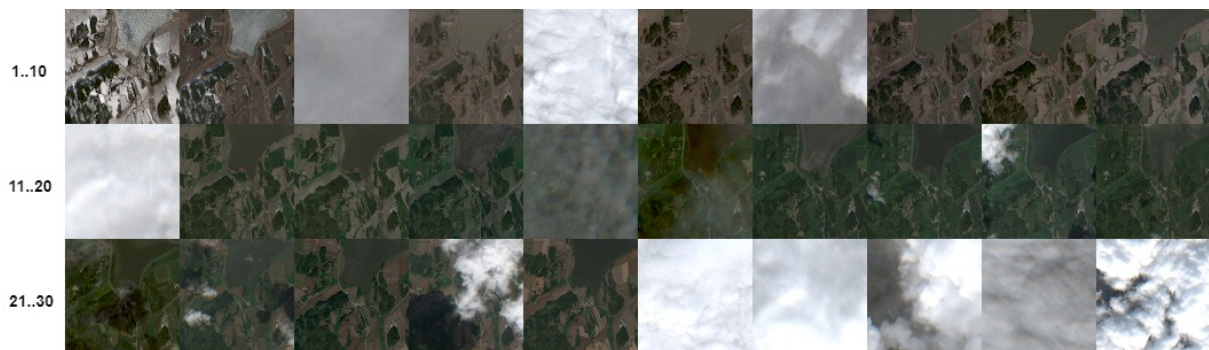


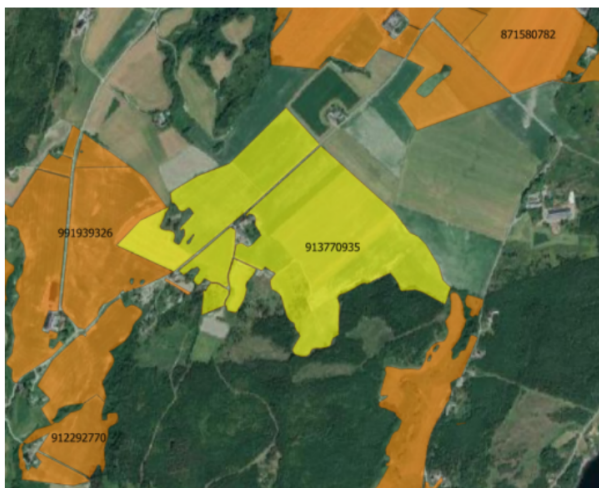
Figure 4.4: Dataset sample showing a 30 week time series of a farm in natural color. Images are indexed 1-30, roughly equal to weeks 10-39.

4.5.1 Generating Masks

As mentioned in Section 4.2.3, intersecting field boundaries from NIBIO with the cadastral boundaries for each farm results in a dataset of precise field boundaries for each farm, and this dataset can be used to generate the masks. The masks are generated in three steps:

1. Do an intersection of the bounding box and the cultivated fields for each farm, leaving only the coordinates for cultivated fields inside the satellite images.
2. Convert the geographic map coordinates of longitude and latitude to corresponding pixel locations of the satellite images. Given that the images are 100x100, the bounding boxes represent the borders of these images (top left 0,0 and bottom right 100,100). Convert each point of a field within the bounding box to pixel coordinates based on relative position within the bounding box.
3. Generate a matrix of zeros and ones based on where the cultivated fields are located, resulting in a 100x100 matrix.

See Figure 4.5 for a visual representation of a mask generated of a farm using the known cultivated fields for this farm, and this mask applied to a satellite image in Figure 4.6.



a) Intersection of cadastral layer and field boundaries of a single farm



b) Visual representation of the resulting mask.

Figure 4.5: From cultivated fields to generated mask. Using the cultivated fields of a single farm (a) to generate a mask (b).

4.5.2 Applying Masks

Once the masks are generated, there are two methods of applying these to the images; applied or added as a channel. In order to apply masks, the mask is multiplied to each channel of the image. The result is an image where only the cultivated fields remain, and all other pixel values are zero. Using the second method, the mask is added onto the image as a separate channel. The mask channel will add some information to the image, so that it can be used to highlight which parts of the image are cultivated fields. See Figures 4.7 and 4.8.



Figure 4.6: Mask applied to an image

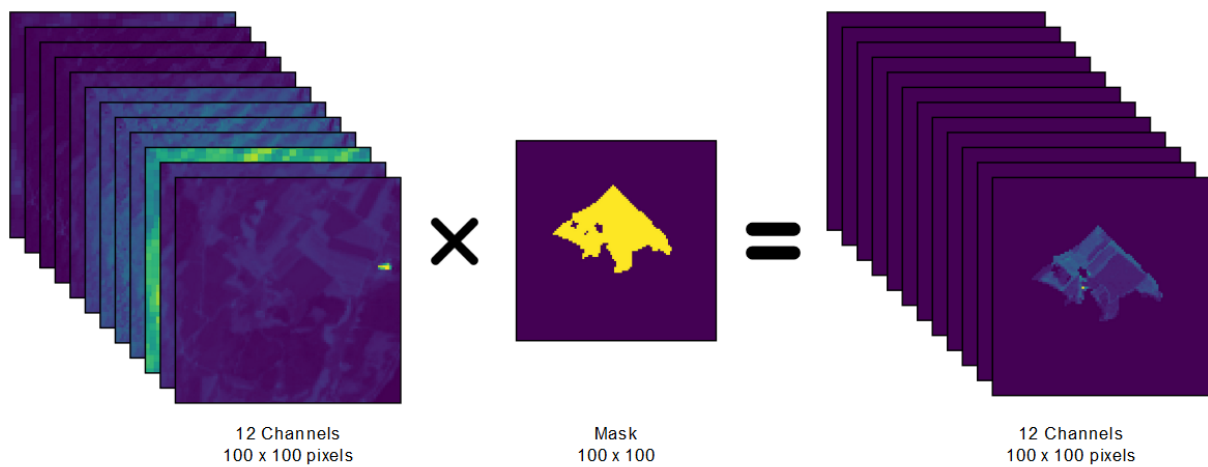


Figure 4.7: Process of applying mask

4.6 Time Spans of Satellite and Weather Data

There are two time series in this dataset: Satellite Images and Weather Data. In Norway, the growing season for grain usually starts mid-April [11], and harvesting is usually done in August, but it can also occur in the period from July until late September, depending on seasonal variations. To be sure that we encapsulate the entire growing season within the dataset, the period between March 1st and October 1st is downloaded for both satellite and weather data.

4.7 Summary

Through all the aforementioned sources and processes, we acquire the dataset for the upcoming experiments. It contains the following features per farm per year for the last three complete production years (2017–2019)¹³: Daily min, mean, and max

¹³To clarify, the primary key for each sample is the farm-year combination. This gives us three samples for a single farm that has produced grain for all three years

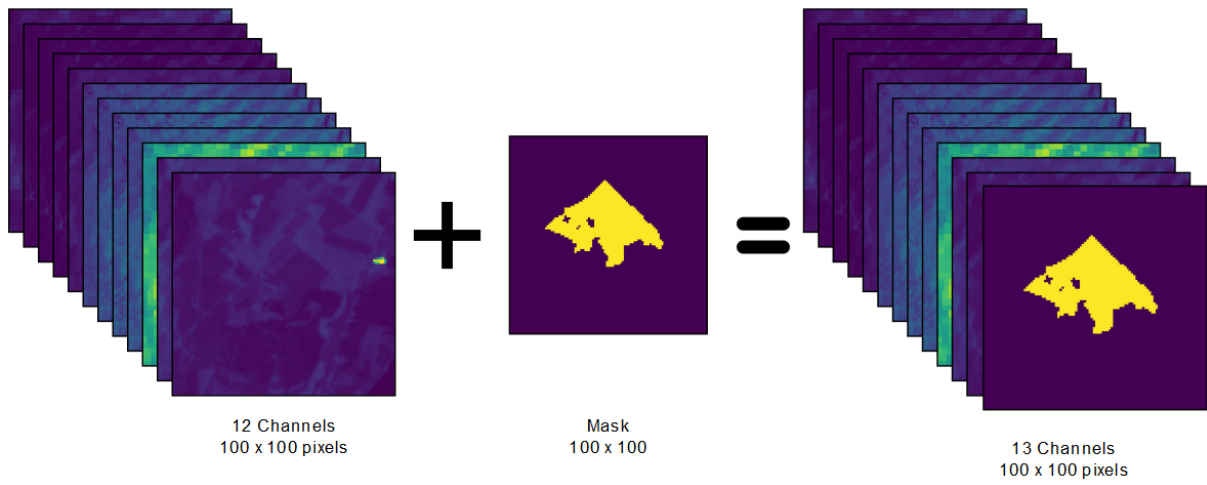


Figure 4.8: Process of adding mask

temperatures; daily total precipitation; latitude; elevation; weekly twelve-band satellite images; field mask; crop yields of the four previous years; harvested area; crop type; and relative crop yields.

Chapter 5

Method

To predict crop yields on a farm-scale basis, we propose multiple prediction models with various capabilities and areas of focus. The general system architecture and approach can be seen in Figure 5.1, where the prediction models use different combinations of the per-farm features to predict the relative yield in kg/daa of each farm. The selection of features is the intersection between features that are known to correlate with crop yield, and the data we have been able to gather. Deep neural networks have successfully been applied to predict average crop yield at different regional scales, suggesting that farm-scale crop yield predictions can be made with the availability of per farm data.

Several deep learning models are implemented to predict yield per decare on a per-farm basis. We present the models in three sections based on what data is processed: Section 5.2 Weather Data includes a model using primarily weather data. Section 5.3 Satellite Images introduces two models focusing on the use of single and multi-temporal satellite images, and Section 5.4 Satellite Images and Weather presents three models using a combination of weather data and satellite images.

In Section 5.1 Data Preprocessing, we explain how the model inputs are preprocessed and how the ground-target values are defined. Lastly, Section 5.5 Reducing Overfitting presents techniques used to expand the dataset and combat overfitting.

5.1 Data Preprocessing

5.1.1 Normalization

Most features are normalized to better fit a range between 0-1, using the linear scaling shown in Equation 5.1. For many features, the upper and lower values are the features minimum and maximum values, resulting in a min-max normalization. For other features, such as the weather features, a fixed normalization range is applied to keep the normalization consistent across the entire time series and measurement aggregations¹. Table 5.1 shows the upper and lower normalization values used in Equation 5.1 for all features which are not min-max normalized. The Sentinel-2 images have pixel values between 0 and 1 from the source, and are therefore not normalized before

¹The temperatures for each day are aggregated by min, mean, and max, as explained in Section 4.3.1.

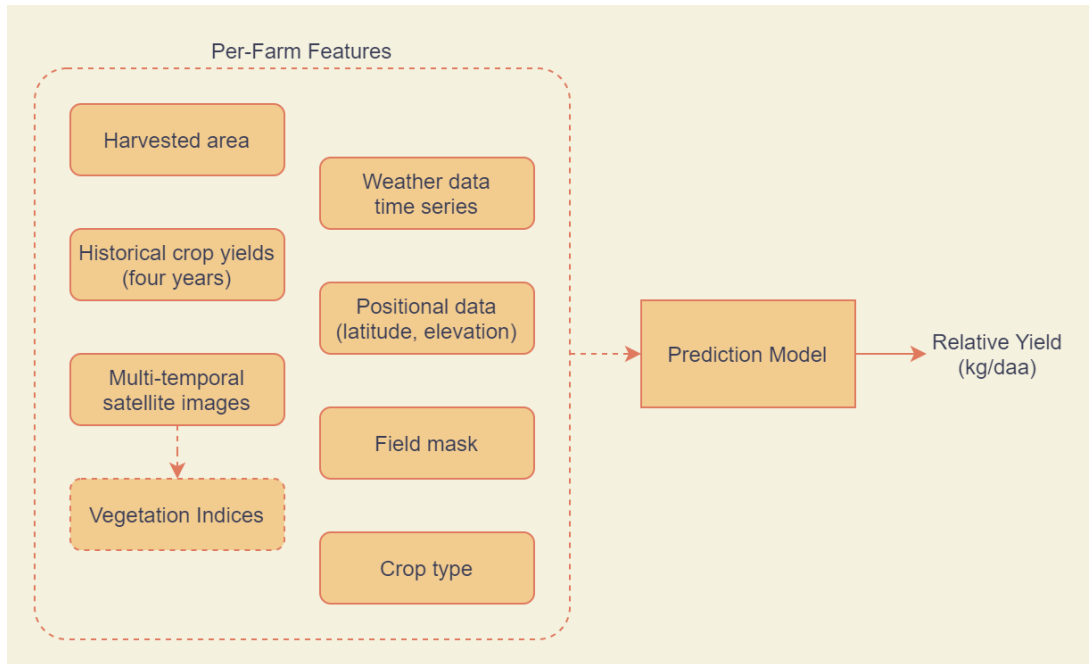


Figure 5.1: Prediction model system architecture

use.

$$\text{normalized} = \frac{\text{value} - \text{lower}}{\text{upper} - \text{lower}} \quad (5.1)$$

Equation 5.1 shows how features are normalized. Lower and upper values are either specified in Table 5.1 or they are set to a feature’s minimum and maximum values. For lower and upper equal to minimum and maximum values, this is called min-max normalization.

| Feature | lower | upper |
|-----------------------|--------------|--------------|
| Crop yield (kg/daa) | 0 | 1000 |
| Temperature (°C) | -30 | 30 |
| Precipitation (mm) | 0 | 10 |
| Historical yield (kg) | 0 | 10 000 |

Table 5.1: The normalization constants (lower and upper) used to scale feature values where min-max normalization was not used.

5.1.2 Prediction Targets

All the proposed models presented in this chapter have a single output, which is the predicted crop yield per decare. However, the target yield is slightly different between some models, requiring some clarification. As the dataset contains how much a farmer has delivered for each crop type, as well as how large areas have been harvested for each type, we can calculate a target yield for each crop type separately as shown in Equation 5.2. Another approach is to use the sum of all crops delivered and the area harvested for each farm, resulting in the total yield per farm as shown in Equation 5.3. The two methods both allow models to be trained to predict farm-scale crop

yield. However, they also produce slightly different distributions, which means that predictions of one type are not directly comparable to the other.

$$y = \frac{\text{Grains delivered (kg)}}{\text{Area harvested (daa)}} \quad (5.2)$$

Equation 5.2 illustrates how the ground truth was calculated for each sample for models that take crop type as input.

$$y_{\text{total}} = \frac{\text{Sum of all grains delivered (kg)}}{\text{Total area harvested (daa)}} \quad (5.3)$$

Equation 5.3 illustrates how the ground truth is calculated for each sample of the Single Image CNN.

The Single Image CNN (see Section 5.3.1) is the only model using the total yield per farm, while all other models are trained on the crop-specific yield targets. As some farmers deliver multiple crop types each year, some samples are duplicated for each of the calculated crop yield targets, and the models are given the crop type as input to differentiate between them. By providing the same image or weather input but with different crop type inputs, the aim is to force the models to learn yield characteristics for each crop type.

5.2 Weather Data

As mentioned in Section 4.3, weather data directly includes information for two of the four main factors of plant growth: precipitation and temperature. Temperature is proven to be highly correlated with yield growth, and precipitation can be useful where the use of irrigation is not widely used, as well as an indicator of drought [17][18]. Training a deep neural network on the weather data allows us to verify the utilization and relevancy of these data.

5.2.1 The Weather DNN Model

A deep neural network from the preliminary project serves as a baseline for our other models [36]. The model is a feed forward neural network consisting of an input layer, three densely connected layers with tanh activation of 512, 128, and 64 units respectively, and one output layer at the end, as shown in Figure 5.2. After the two hidden layers there are dropouts of 10 and 25 percent, as that seems to give the best generalization.

The model has 883 input features, 856 (96.9%) of which are weather features. The remainder are historical, positional and other relevant features such as the cultivated area and crop type.

5.3 Satellite Images

Satellite images are remotely sensed data collected by earth-observing satellites. These images are available globally, cheap, and include detailed high-resolution observations of the earth. The multispectral satellite images contains detailed information about

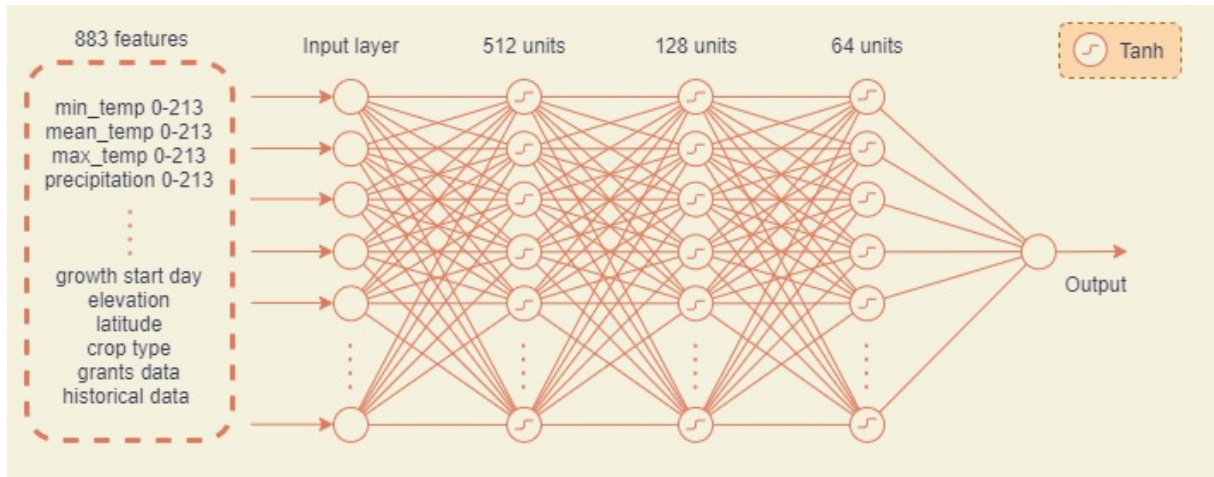


Figure 5.2: Weather DNN architecture

crop growth and plant health, which traditionally have been extracted using hand-crafted vegetation indices. By training on per-farm satellite images, the aim is that models are able to automatically extract relevant features that are important for crop yield. The models discussed in the upcoming subsections all use raw satellite images to extract relevant features and make crop yield predictions based on these.

5.3.1 The Single Image CNN Model

The initial model using satellite images was a simple CNN. This model aims to act as a proof of concept for the thesis and indicates whether satellite images of farms in Norway contain some information that can be used with deep learning to predict grain yield.

To keep the model as simple as possible, it takes *one* multispectral image as input and makes yield predictions based on this. The ground truth is calculated by summing all grain deliveries and dividing this by the total area harvested for each farm, resulting in a grain yield target specified in kg/daa, and can be seen in Equation 5.3.

The model input layer is by default $100 \times 100 \times 12$, meaning images of 100×100 pixels and 12 channels deep. In specific experiments, such as adding mask as channel and cropping the images, the input layer is adjusted slightly to accommodate images of size 90×90 or with 13 instead of 12 channels. The CNN layers are made out of three pairs of 2D convolutional layers and 2D max-pooling layers. Each pair has an increasing number of 3×3 convolutional filters with ReLu-activation applied to the input (16, 32, and 64, respectively) and a max-pooling of size 2×2 reduces the output dimensions at each step. Next, the last max-pooling output is flattened before being fed into a dense layer of 32 units with a single dense layer at the end. See Figure 5.3.

5.3.2 A Multi-Temporal CNN-RNN Model

As examined by Jiang et al., the developments of crop phenology play an essential role in the eventual harvested yield. Grain crops are typically planted either during autumn or in the spring, and the growth progress from seedlings to mature harvestable crops are not constant or fixed through time [25]. By training a model on multi-temporal

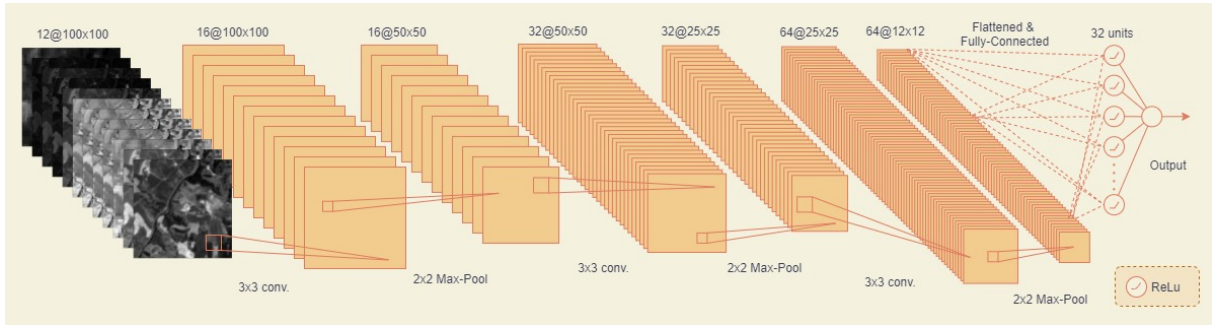


Figure 5.3: Single Image CNN Model architecture

images from the growing seasons, the aim was to achieve higher accuracy by also analysing the changes that occur over time.

The proposed model is a convolutional and recurrent neural network that takes an image time-series as input, and outputs the predicted yield/daa. The model is illustrated in Figure 5.4, and uses a similar CNN architecture to the Single Image CNN model (Figure 5.3) for each image. The CNN output is then fed into a GRU encoder network together with a one-hot encoding of the crop type and a normalized field area which is duplicated for each timestep.

The architecture of the CNN used for each timestep has the same convolutional and max-pooling layers as the Single Image CNN, but the two dense layers are replaced by a single 64 unit fully-connected output-layer, effectively reducing each image down to a 64 element vector. The same CNN weights are reused for each timestep, meaning that the network size is independent of sequence length. The CNN output is concatenated with crop type and area, and then fed into a GRU encoder with 128 units. The GRU encoder output is fed through a single fully-connected layer with ReLu activation, and the output is a single neuron with no activation function (linear).

5.4 Satellite Images and Weather

Between the Weather DNN and Multi-Temporal CNN-RNN, all of the previously described data is used as feature inputs. However, the models train and predict individually, which means that the models cannot learn any patterns that only appear when both satellite and weather data are combined. For this reason, one LSTM and two hybrid models were created. For the LSTM model, the satellite images are condensed into four vegetation indices before being used as input to the model. The hybrid models combine different architectures so that weather and raw satellite images can be used in conjunction to provide better-informed predictions.

5.4.1 Handcrafted Features in LSTM

Following the work of Johnson and Bendetti [17][4], the purpose of this model is to evaluate the use of vegetation indices and weather data to predict yield on a farm scale. Additionally, by using vegetation indices directly in a time series model, we can compare these results to how well the models using raw images performs.

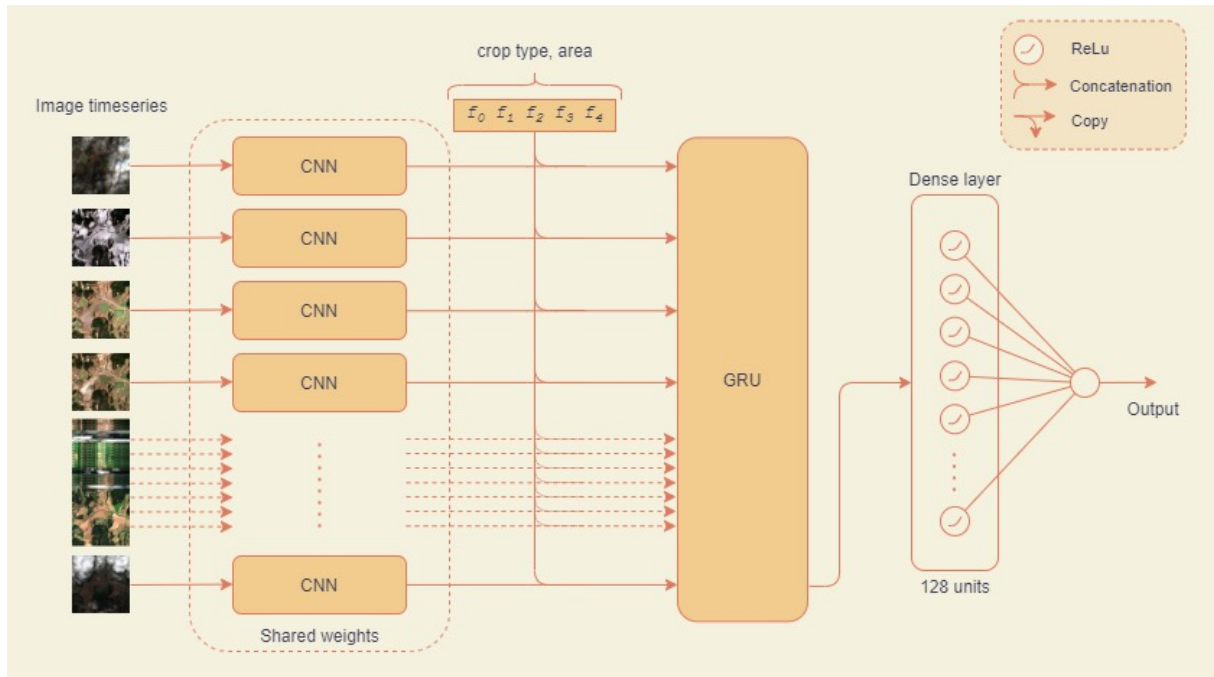


Figure 5.4: Multi-Temporal CNN-RNN Model architecture

Handcrafted Features

As mentioned in Section 4.4.2, the Sentinel-2 time series data is collected in 7-day intervals, and the weather features has to be ingested to synchronize with these intervals. The temperature and precipitation measurements are grouped in 7-day intervals, such that the time window matches with the Sentinel-2 data. Next, the temperature features of min, max, and mean are calculated for each group, resulting in 3 temperature features per interval. The precipitation is summed for each group, resulting in one precipitation feature for the total precipitation per interval.

Next, the vegetation indices are calculated. Image masks are applied to each image so that only cultivated crops remain (See Section 4.5 for details of masking). Much like Johnson and Bendetti, each of the indices are calculated for all remaining pixels, and the mean value represents the actual vegetation index. The specific vegetation indices include NDVI, WDRVI, NDWI, and NDMI (See Table 5.2 for details), resulting in 4 vegetation indices per interval.

Alongside the growing seasons time-series data, each sample also includes crop type, historical yield, area utilized, latitude, and elevation data. This results in 30 (intervals) x 8 (4 vegetation indices and 4 weather) timestep features, plus 5 additional features encoded in an input vector of size 22, as seen in Figure 5.5.

LSTM Model Overview

The model used for this is a simple LSTM based architecture, as seen in Figure 5.5. The LSTM section encodes the time-series-based data across the growing season. Next, the encoded growing season data is concatenated with the additional farm-related properties in dense layers. The final output is the predicted yield in kg/daa.

The LSTM consists of 30 cells (timesteps), where each of these cells contains 32 units.

| Name | Abbr. | Formula | Description |
|----------------------------------------|-------|---------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Normalized Difference Vegetation Index | NDVI | $\frac{B8 - B4}{B8 + B4}$ | Indicator of green leaf area, giving a measurement of healthy green vegetation in any given pixel. First used by Rouse et. al. [30] |
| Wide Dynamic Range Vegetation Index | WDRVI | $\frac{\alpha * B8 - B4}{\alpha * B8 + B4}$ | Modification of the NDVI with an extra weighting coefficient parameter. Increased sensitivity when areas with moderate to high biomass are investigated. [10] |
| Normalized Difference Water Index | NDWI | $\frac{B3 - B8}{B3 + B8}$ | A measurement that is sensitive to changes in water content of vegetation. [8] |
| Normalized Difference Moisture Index | NDMI | $\frac{B8 - B11}{B8 + B11}$ | Indicator of the water content of vegetation. Effectively similar to that of NDWI, but calculated using other aspects of the spectrum. [8] |

Table 5.2: Handcrafted features used as input. The bands (B) are sentinel-2 specific, see Table 4.3 for details.

After the LSTM layer, there is a dropout layer of 25 percent, a dense layer of 32 units, and another dropout layer of 10 percent. Next, the farm-related properties input layers are concatenated into the network before a 32 unit dense layer and a final single neuron output. The LSTM units uses the activation function tanh, and all dense layers use ReLu.

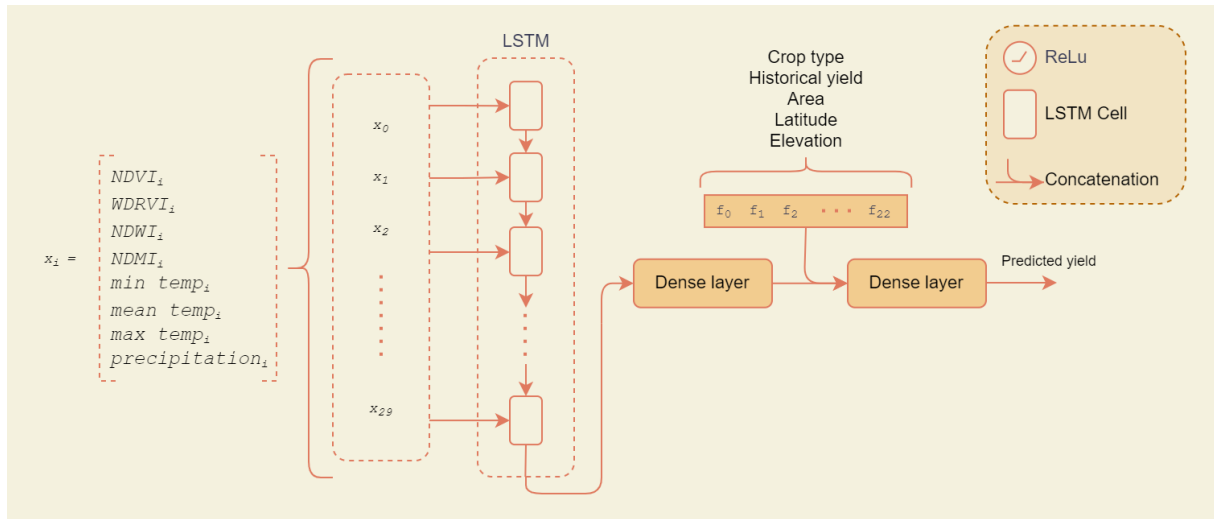


Figure 5.5: Handcrafted features in LSTM architecture

5.4.2 Hybrid 1: Pre-Trained Hybrid Model

The Pre-Trained Hybrid Model (Figure 5.6) combines the Weather DNN and Multi-Temporal CNN-RNN by concatenating the outputs of the second to last layers and feeding it into a deep neural network consisting of three fully connected layers. The

first two layers of the combined network use ReLu activation and have 64 neurons each, followed by 10 percent dropouts, and the last layer of the combined network is a single neuron that outputs the predicted crop yield. The Weather DNN and Multi-Temporal CNN-RNN are trained separately, first the Multi-Temporal CNN-RNN, followed by the Weather DNN. Then, all layers before (but not including) the second to last layers of the pre-trained models are locked so that their weights do not change. Finally, the complete hybrid model is trained, combining the value of both sub-networks.

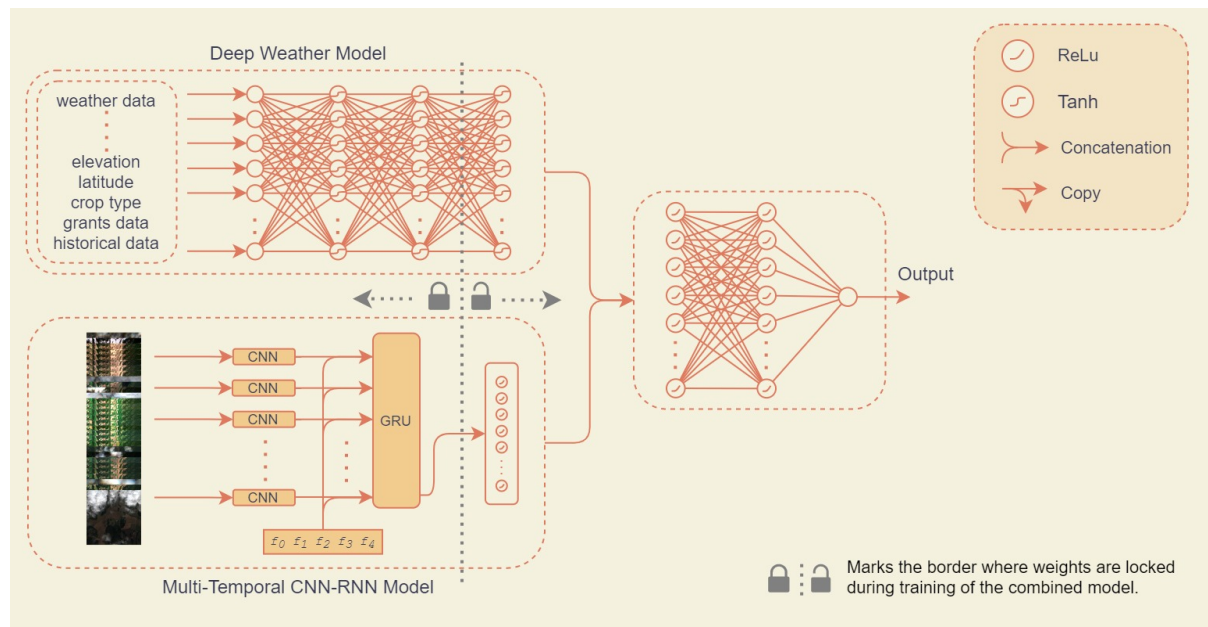


Figure 5.6: Pre-trained Hybrid Model architecture

To avoid leaking validation data into training data between any of the three stages, the dataset is prepared so that the training/validation split would remain constant through all stages. Because the weather data has significantly more samples that increase error when excluded, custom separate dataset generators provide each stage with all relevant features and samples while keeping the training/validation split the same for all three.

5.4.3 Hybrid 2: Hybrid CNN Model

Although the pre-trained Hybrid Model combines all the available features and data for each farm to make predictions, the method makes the model slightly cumbersome to train, and the architecture is inefficient as some of the learning in the two individual models are simply discarded when the last layers from both models is skipped in the combined model.

The second proposed hybrid model, shown in Figure 5.7, is trained in a single stage and has fewer trainable weights than the first, as well as combining satellite and weather data in 7-day time steps that allow the model to process a single sequence using both data sources. The model shares much of the architecture with the Multi-Temporal CNN-RNN Model, but also incorporates weather data through a 1-dimensional convolution which allows both satellite and weather data to be encoded in 7-day time steps. The model then combines the output from the encoded satellite images and the encoded 7-day weather data by concatenating both vectors for each timestep. The con-

concatenated vectors are then fed into a GRU encoder which encodes the whole sequence into a 128 length vector, and a fully-connected layer with ReLU activation along with a single output neuron predicts the crop yield per decare.

Both weather and satellite data are captured from the 1st of March to the 1st of October for each sample, however, satellite images have a temporal resolution of 7 days, while the weather data has a higher temporal resolution of 1 day. To combine these two data sources, we apply a one-dimensional convolutional layer, with size and stride of 7, on the weather inputs. The size and the stride used in the one-dimensional convolution means that the weather time series is reduced down to 30 vectors, effectively encoded as 7-day intervals. The 1-dimensional convolutional layer has 64 filters, meaning a 7-day period of weather data is encoded as a vector of length 64, the same size and temporal resolution as the CNN outputs from the satellite images sequence.

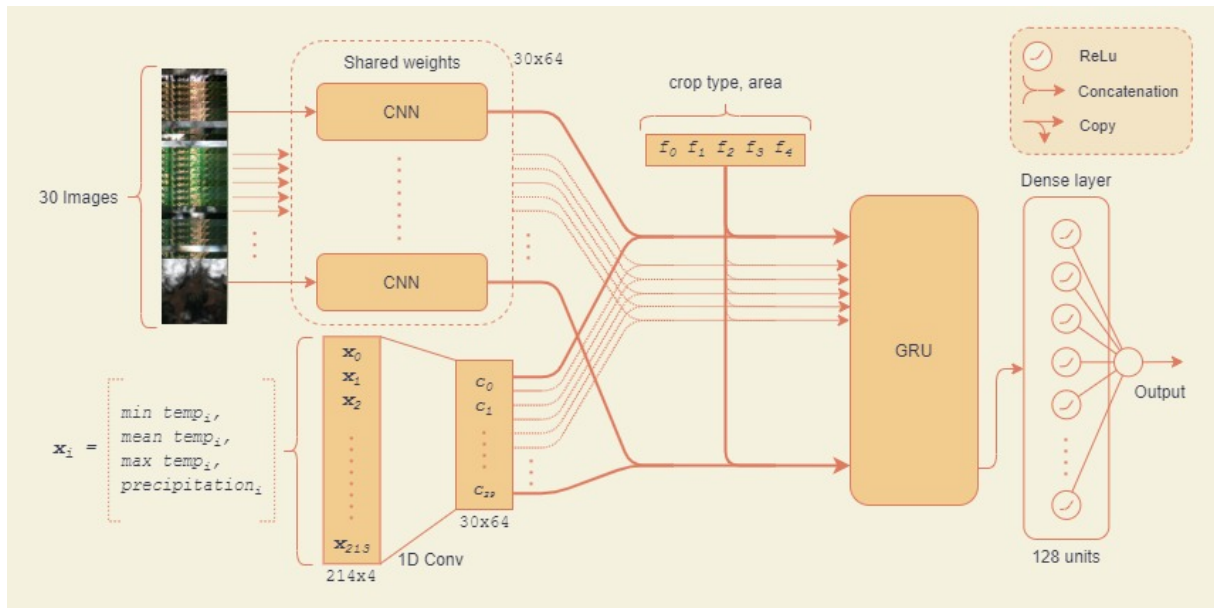


Figure 5.7: Hybrid CNN Model architecture

5.5 Reducing Overfitting

The models included in this thesis are deep learning-based, which can be considerably data-hungry. The dataset involved spans three years (2017, 2018, and 2019) and consists of 509 910 unique images, whereas one sample contains 30 images, resulting in 16 997 samples in total. Overfitting was observed in all the models which train on raw satellite images, and we therefore extensively used data augmentation techniques on the images to increase the overall dataset size and to combat overfitting. We also apply a stochastic epoch sampling technique which allow us to stop training when the models start overfitting and restore the best weights.

5.5.1 Data Augmentation

Data augmentation is primarily used for the CNN-based models taking raw satellite images as input as these tend to overfit when training, and data augmentation techniques allow us to generate variability on the images to prevent this. We implement

three main data augmentation techniques: image cropping, image rotation, and random pixel noise. Because the memory requirements of the complete dataset is too large to fit in GPU memory, or even RAM, images are continuously read from storage while the model is training on the previous batch. The data augmentations are applied to images only after they are read from storage, and because both rotation and noise is performed with some randomness, a complete cycle of the augmented dataset is never the same. No data augmentation was performed on the validation samples. However, for models that take cropped images, only the center crop was used for validation (as seen in Figure 5.8).

Cropping

The cropping augmentation is a method to extend the dataset. Initially, the images are 100x100 pixels, and by cropping these to 90x90 pixels, we extend the dataset by a factor of five, with minimal loss of information. Each training sample is cropped five times, such that the resulting dataset has five entries for the same farm. The crops are done top-left, top-right, center, bottom-left, and bottom-right. See Figure 5.8 for a visual representation.



Figure 5.8: Cropping augmentation visualized in true color

Rotation

Another common method of increasing the number of training samples is to apply image rotation, which forces the models to learn features not purely based on the location of certain specific patterns in an image. We extensively apply image rotation to all image samples by selecting a random rotation angle for each image. The angle of rotation is also different for images in the same time series, which produces a unique image time series every time, illustrated in Figure 5.9. Image dimensions are kept unchanged, meaning that any corners rotated out of the image are discarded. Black pixels are used to pad any empty corners of the image.

Pixel Noise

To increase variability of images even further, we introduce some augmentation through noise, using a simple salt-and-pepper method. Applying salt-and-pepper noise is a process of changing a fraction of the pixels in the image to their minimum or maximum values (0 or 1) [40]. The vast majority of the image remains unchanged, while approximately 1 percent of the pixels (chosen at random) were altered to either 0 or 1. Since the images in the dataset are multispectral (i.e., containing 12 channels), when a pixel was chosen to be altered, the value of all the channels was updated at

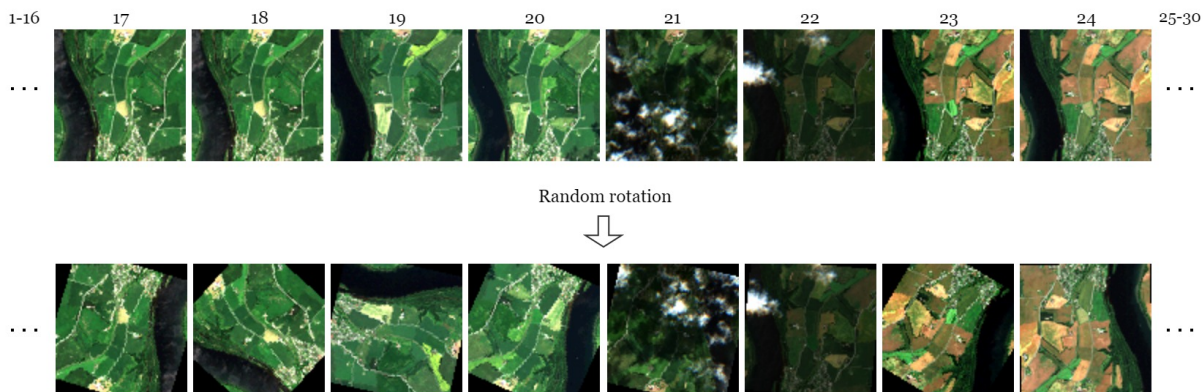


Figure 5.9: Rotation augmentation visualized in true color

the same pixel. See Figure 5.10 for a visual representation of salt-and-pepper applied to a satellite image.

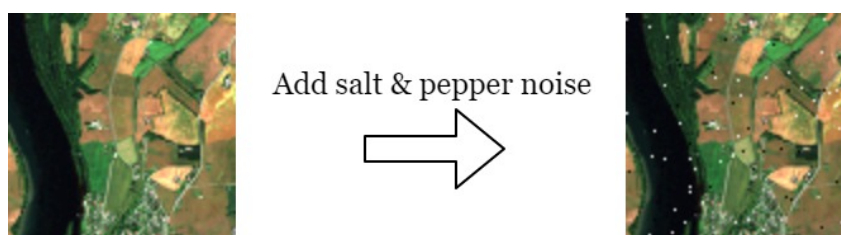


Figure 5.10: Salt-and-pepper augmentation visualized in true color

5.5.2 Stochastic Epoch Sampling

For the models that are trained with satellite image time series, the size of each training sample (without cropping) is at least $100 \times 100 \times 12 \times 30 = 3600000$ parameters, when not counting additional inputs to the models. We suspect the large number of parameters for each training sample is the reason the models are prone to overfitting, as an increased number of dimensions will eventually include unimportant data, which makes it harder to generalize well on the important features[12]. The size of training samples also slows down the training speed, as much more data must be read from storage for each sample. The observed training speed varies between systems with different GPUs and storage options, however, a minimum of 5 hours training time per epoch with the augmented dataset would be expected using the available hardware. During this time, the model might have already started to overfit on the training samples, but without testing on validation samples more frequent, it is impossible to know exactly when the model performed best.

Validation is typically done at the end of each epoch, and because data augmentations increases the dataset size and thus the number of samples in an epoch, the time between validations also increases. We solve this by artificially reducing the size of each epoch by taking a random subset of the samples instead of the whole set. This leads to much more frequent validation runs, which allow us to better monitor how well the model is learning. Figure 5.11 illustrates how stochastic epoch sampling provides higher resolution on the monitored loss values. In all our experiments, training is stopped when validation loss has not improved for a set number of epochs, and the

model weights are restored from the epoch with the best validation loss. The stochastic epoch sampling ensures that we can restore the model weights from the best point in time, while also preventing models from training unnecessarily long.

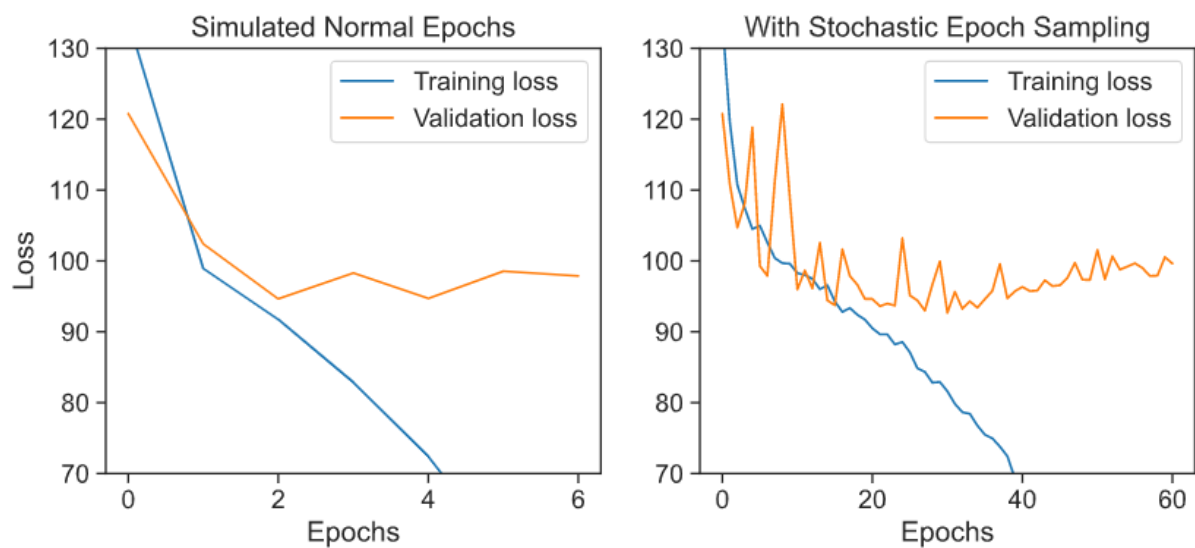


Figure 5.11: Illustration of stochastic epoch sampling

Chapter 6

Experiments and Results

This chapter presents the experiments and results of the thesis. The experiments conducted in Section 6.1 are primarily intended to assess the use of satellite images as a valid data source and to further quantify the use of data augmentation methods and image masks. Section 6.2 shows the performance and comparisons of all models described in Chapter 5. Section 6.3 describes the capabilities of the best performing model to make early, in-season predictions and Section 6.4 scales the model up to predict on a per commune basis, so that a comparison can be made between the proposed model in this thesis, and the work of Sharma et al. [35], presented in Chapter 3.

6.1 Initial Experiments on Multispectral Images

This section includes experiments conducted to evaluate the use of multispectral imagery to predict yield, give indications of the effectiveness of data augmentation methods mentioned in Section 5.5.1, and test the use of image masking discussed in Section 4.5.

The experiments use the Single Image CNN, which is built to be a simple CNN-based model so that the impact of the different experiments becomes clear. As explained in Section 5.1.2, this model does not differentiate between crop types, and was trained to predict total crop yield per decare for each farm. This means that the mean absolute error (MAE) cannot be directly compared to that of the other models in the thesis.

6.1.1 Evaluating the Use of Multispectral Imagery for Yield Prediction

The first experiment conducted with the multispectral images checks whether or not a CNN can predict the yield with better accuracy than if one just predicted the average. By calculating the y_{total} (all grains delivered in kg divided by total area harvested) per farm, the mean yield per decare across the training dataset results in 381.2 kg/daa. When using the mean yield per decare as the prediction on the validation data, the MAE is 134.3 kg/daa. Anything lower than this means that the model has found some relevant features from the multispectral images.

The Single Image CNN is trained using one image per year for each farm, which results in a validation loss of 96.4 kg/daa MAE, as seen in Figure 6.1. This indicates that the

model is able to learn relevant features from the multispectral images, validating the use and further exploration of satellite images.

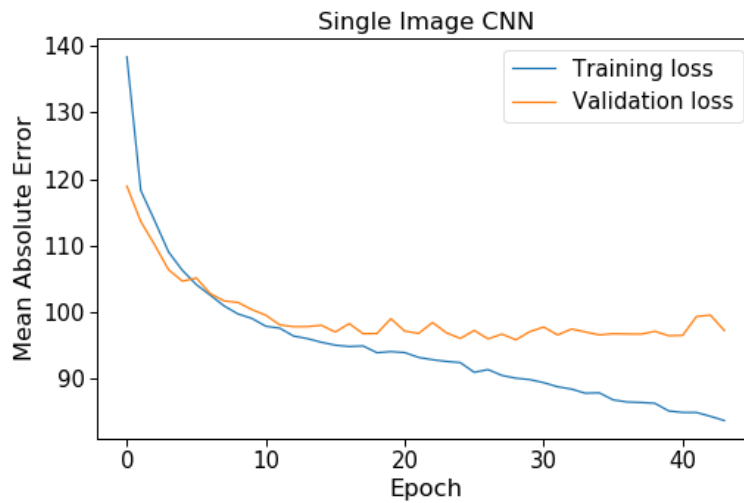


Figure 6.1: The initial results of the Single Image CNN model

6.1.2 Optimal Week to Predict Yield

The continued development and stages of crop growth play an important role in the harvested yield, which could mean that selecting different weeks as input for the Single Image CNN will increase or decrease the performance. To determine which week of the chosen period (March 1st to October 1st for each year) contains the most relevant information, the model is trained separately for each week.

As can be seen in Figure 6.2, there are measurable variations in the performance across the growing season. The period from week 26 to 29 seems to be the weeks where the model performs the best, which corresponds to roughly the period from 25th of June to 22nd of July. Interestingly, this finding is similar to what Basnyat et al. found to be the optimal time to use remote sensing for crop grain yield on the Canadian prairies, which was between the 10th to the 30th of July [3]. Based on these findings, week 26 is chosen for all single-image based experiments going forwards.

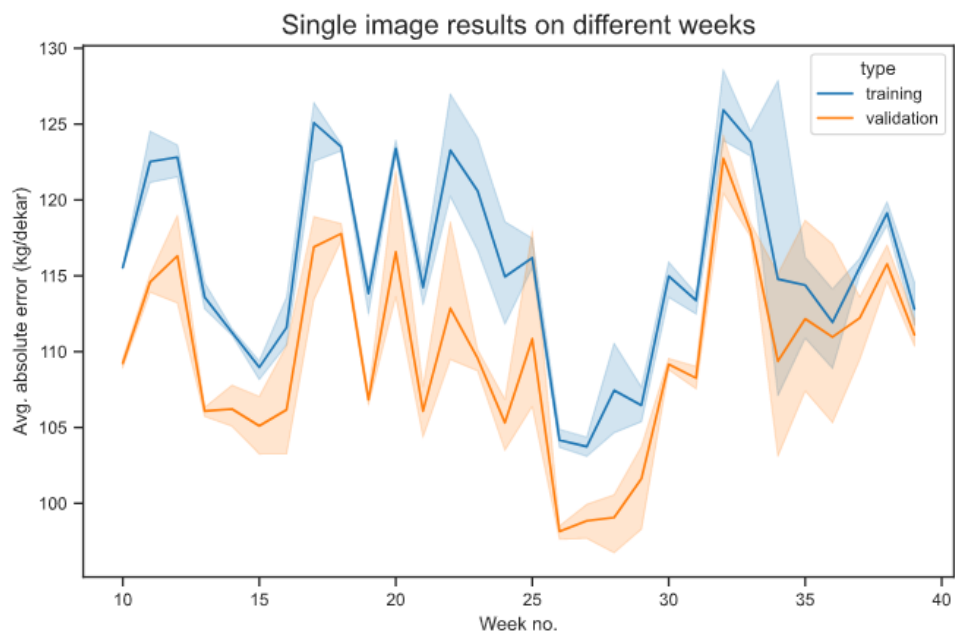


Figure 6.2: Weekly performance comparison of the Single Image CNN. The model was trained separately for each of the weeks, limited to 10 epochs per run, and each run was performed three times. The limited number of epochs is why validation loss is typically lower than the training loss in these experiments.

6.1.3 Effects of Data Augmentation

As can be seen in Figure 6.1, the model shows signs of overfitting on the initial runs, hence we add data augmentation as an effort to reduce overfitting and increase the overall performance. The data augmentation methods used are cropping, rotating, and adding noise (More details in Section 5.5.1).

The datasets are split into training and validation sets before augmenting the training data. When cropping, the validation images also have to be cropped to match the model requirements of 90x90 images, and the centered 90x90 crop is applied. For these specific experiments, the rotations are 90°, 180°, and 270°.

Each of the augmentation methods are tested separately. The results for each of the methods consist of three runs, and can be seen in Figure 6.3 and Figure 6.4. There are improvements both in regards to less overfitting, as well as lower loss overall.

By evaluating the best-achieved loss, as shown in Table 6.1 and Figure 6.5, we see that the salt-and-pepper brings a modest improvement of 3.3%, while the improvements from cropping and rotating are both at about 7.1%. Additionally, when combining rotating, cropping and salt-and-pepper, the validation loss further improves to 10.1% overall.

These positive results of augmenting the satellite images suggest that the models would improve as more years of data becomes available.

| Augmentation | Validation Loss (Mean) | Improvement Over Original (Percent) |
|---------------------------------------|-------------------------------|--------------------------------------------|
| Original (No augmentation) | 89.8 | |
| Salt-and-pepper noise | 86.8 | 3.34 % |
| Rotating | 83.4 | 7.13 % |
| Cropping | 83.3 | 7.24 % |
| Salt-and-pepper + Rotating + Cropping | 80.7 | 10.13 % |

Table 6.1: Effects of the augmentations. The validation loss is the mean of three separate runs.

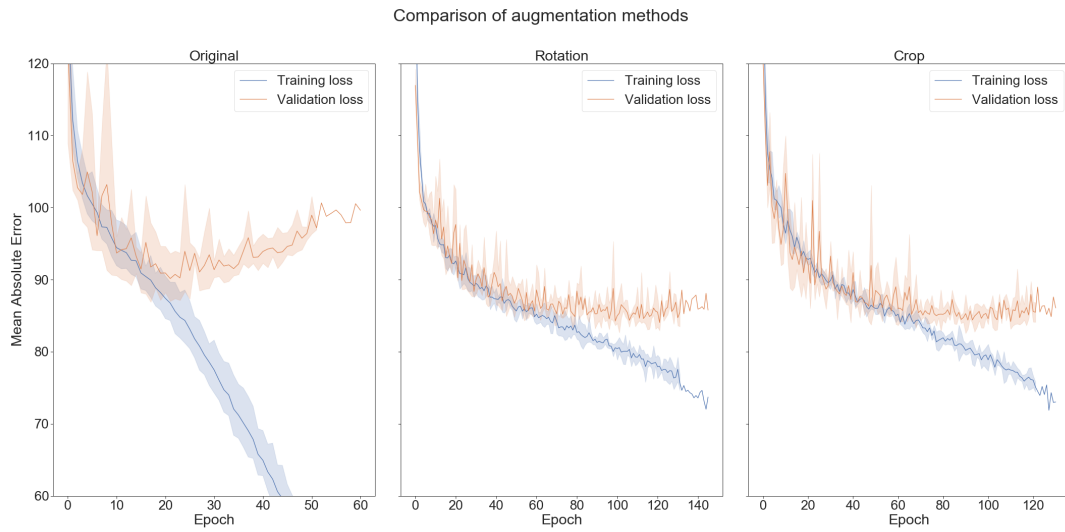


Figure 6.3: Training and validation loss of cropping and rotating

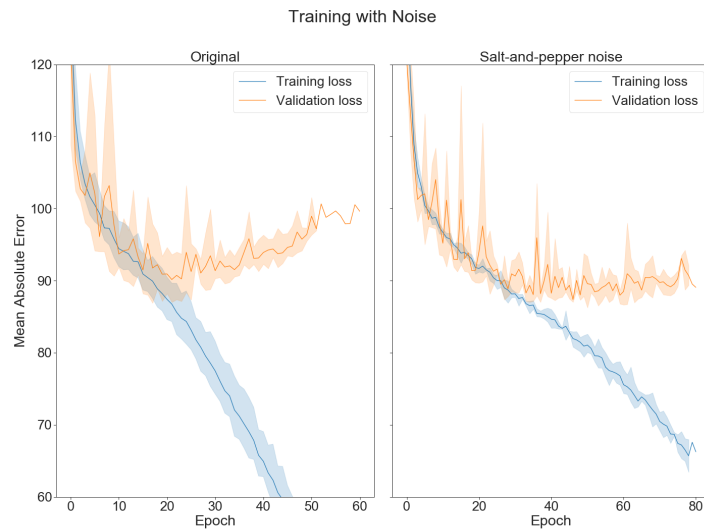


Figure 6.4: Training and validation loss of applied salt-and-pepper noise

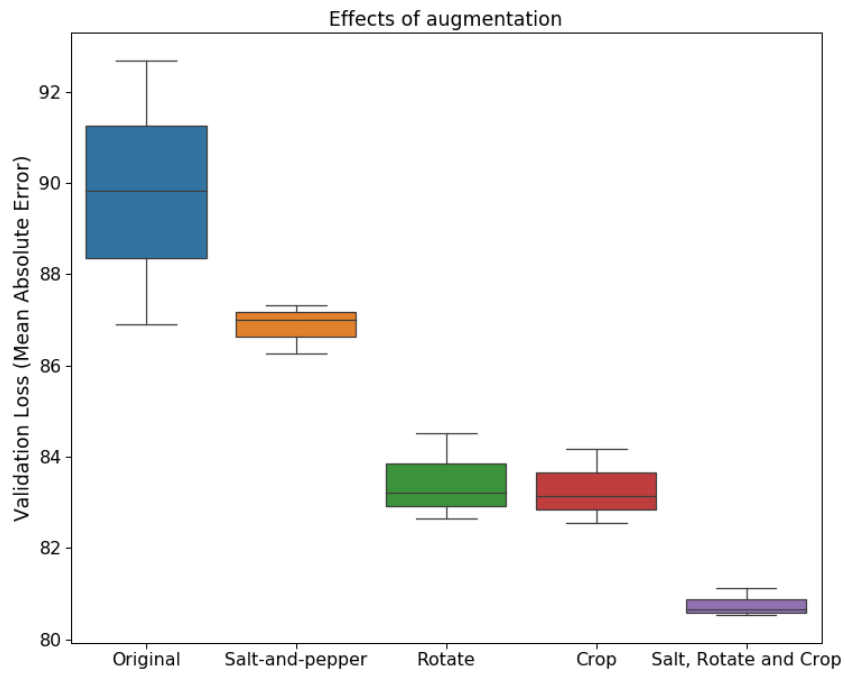


Figure 6.5: The achieved mean absolute error across three runs for each type of augmentation. The original is the baseline with no augmentation applied. Salt-and-pepper provides a modest improvement in performance, however more consistent. Both augmentation methods of rotation and cropping have good effects. Combining all three methods provides even better results.

6.1.4 Effects of Masking

By quantifying the effects of masking, we can attempt to answer the second hypothesis. The second hypothesis questions whether accurate field boundaries can increase the accuracy significantly. In this context, field boundaries along with the satellite images, can be applied using pixel masks. In theory, the masks should remove or highlight the cultivated crops and enable the models to focus on crop-specific features. The purpose of this experiment is to test the use of image masks and the two proposed methods of applying these, as explained in more detail in Section 4.5.2.

Figure 6.6 shows the results of both masking techniques compares to the original without any mask. Both applied and as a channel seems to aid the model and improves the validation loss achieved. Surprisingly, masks as a channel perform the best by quite a margin. The reason for this could be that adding masks as a channel primarily adds information to the image in a separate channel, which suggests that information of the environment around and close to the cultivated fields is also relevant.

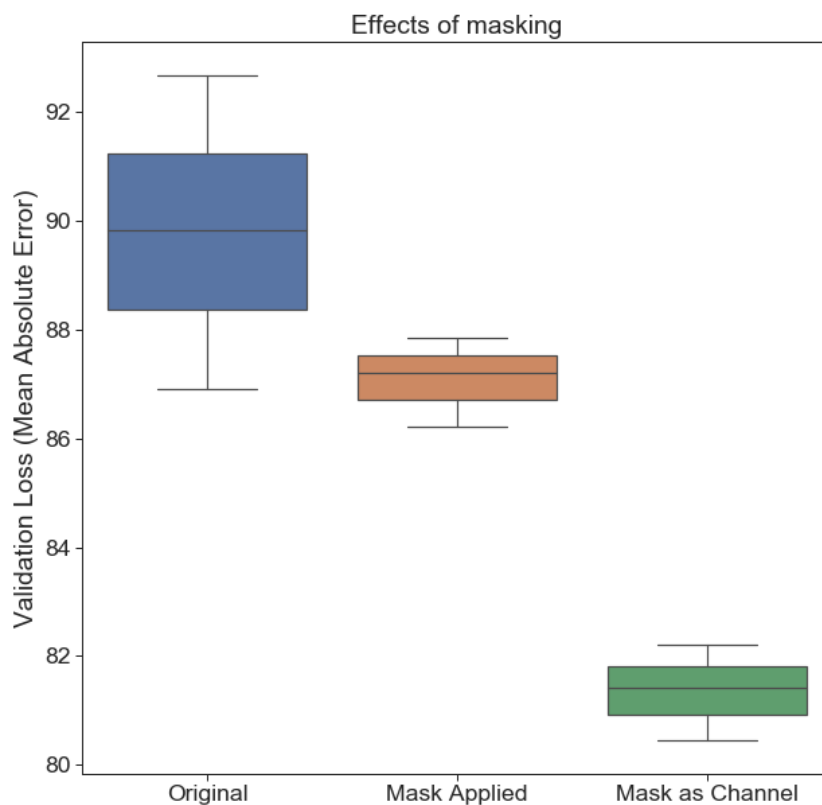


Figure 6.6: The effects of masking techniques on MAE

6.2 Crop Yield Model Comparisons

We evaluate the proposed models by comparing the achieved mean absolute error in Table 6.2, which show that the two best performing models, the two Hybrid models, incorporate both weather data and satellite images. The Single Image CNN model is left out of this comparison, as its prediction targets were total yield per decare where all other models predict crop yield per decare for each of the crop types individually, which belong to different distributions and are difficult to compare directly.

| Model | Mean absolute error (kg/daa) |
|------------------------------|-------------------------------------|
| Weather DNN | 83.04 |
| Multi-Temporal CNN-RNN | 80.52 |
| Handcrafted features in LSTM | 82.29 |
| Hybrid 1: Pre-Trained Hybrid | 77.53 |
| Hybrid 2: Hybrid CNN | 76.27 |

Table 6.2: Best mean absolute error achieved for each model

The baseline model, the Weather DNN model, achieves a mean absolute error of 83.04 kg/daa using daily interpolated temperature and precipitation values as described in Section 4.3.2. The interpolated weather data results in an improvement of around 10 kg/daa compared to previous results using only measurements from the nearest weather station[36].

The Weather DNN results represent the benchmark figure for which we test our other models on the first and third hypotheses: whether satellite images can be used to predict yield accurately and if satellite images combined with weather data increase the accuracy further. The Multi-Temporal CNN-RNN clearly validates the first hypothesis, and the two hybrid models show that combining satellite images and weather data is also beneficial. The second hypothesis, concerning accurate field boundaries, are tested by running the best performing model, the Hybrid CNN, both with and without masks, showing that masking of fields improves accuracy (see Table 6.3). Further results and insights from the other models are presented and discussed in the following sections.

6.2.1 Multi-Temporal CNN-RNN

The Multi-Temporal CNN-RNN model utilizes the satellite image dataset to the fullest, by processes all the 30 images for each sample to give a yield prediction. The model achieves a mean absolute error of just above 80 kg/daa, an improvement from the Weather DNN. However, compared with the Weather DNN, the Multi-temporal CNN-RNN is not given a farm’s previous grain deliveries or positional data; the model is trained using only multispectral satellite images and the crop type and area encoding. Figure 6.7 shows the training and validation loss for the Multi-Temporal CNN-RNN model. The training was performed using images with pixel mask added as a separate channel, as that gave the best results in the Single Image CNN experiments.

While a multispectral satellite image contains a lot of information compared to, say, a temperature and precipitation measurement, a major drawback is the occurrences of cloudy images, which can sometimes constitute a large portion of the images in the 30 image time series samples. The model still manages to predict the crop yield

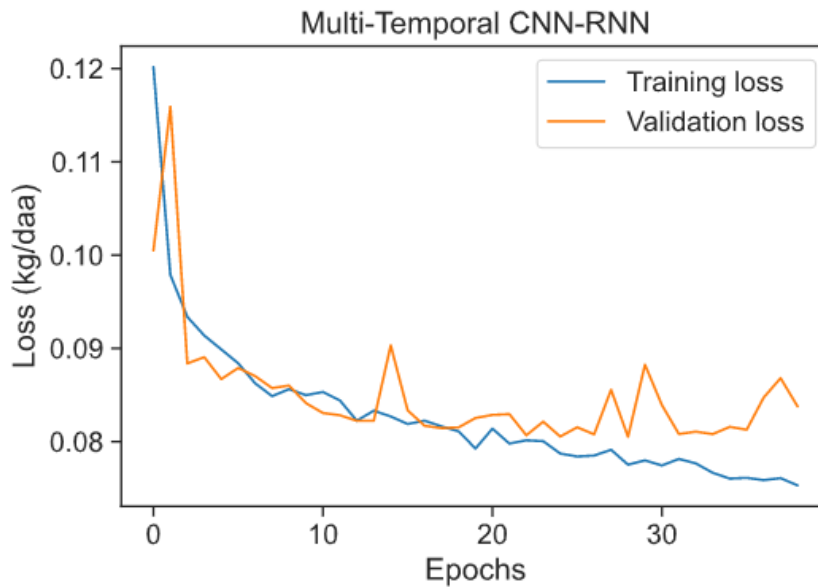


Figure 6.7: Training and validation loss for the Multi-temporal CNN-RNN model

more accurately than the Weather DNN, suggesting that it is able to successfully extract valuable information from the good images while ignoring the noise generated by cloudy images. This is perhaps due to the GRU-encoders ability to control how much each input should contribute to the encoding at each timestep with the update gate (GRU explained in Section 2.1.4).

6.2.2 Handcrafted Features in LSTM

By condensing the relevant sequential time series data into sequences of vectors suited for an LSTM, the LSTM trained on handcrafted features achieves a mean absolute error of 82.29 kg/daa. The model is tested with three sets of sequential inputs: Weather, vegetation indices, and a combination of weather and vegetation indices. Surprisingly, using weather alone only results in a MAE of 93.63 kg/daa, while vegetation indices see a MAE of 83.01 kg/daa. With both weather and vegetation indices, the MAE improves to 82.29 kg/daa. See Figure 6.8 for a comparison of the training.

The relatively poor MAE results of weather features alone can indicate that weather features condensed into four measurements per week removes important granularity. Overall, the MAE achieved with this model is a modest improvement over the Weather DNNs results. One contributing factor to this result can be that although we have information on which fields the farmers cultivate each year, we cannot differentiate the type of crops in the fields. By not being able to make vegetation indices for just the relevant types of crops, they may be affected by fields of grass, potatoes, vegetables, etc.

6.2.3 Hybrid 1: Pre-Trained Hybrid

The LSTM model with handcrafted features shows that combining vegetation indices derived from satellite data with weather data only improves predictions marginally. On the contrary, the Pre-Trained Hybrid model shows that combining the full Weather DNN and the Multi-Temporal CNN-RNN, a definite improvement is achieved compared

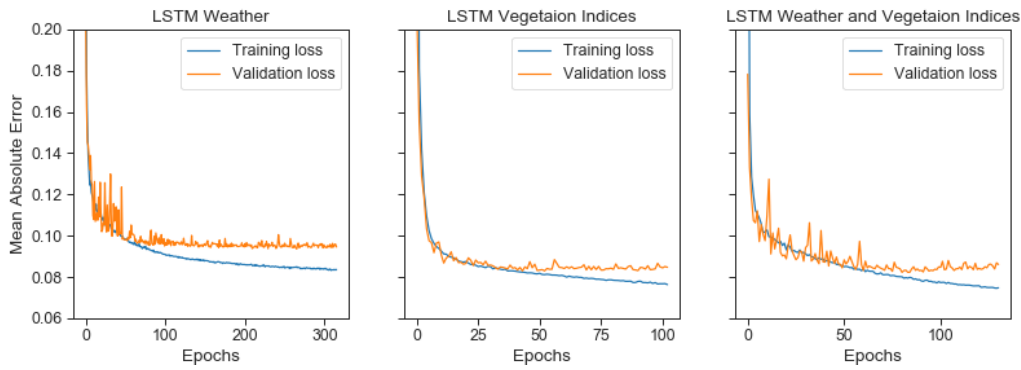


Figure 6.8: Training and validation loss for the LSTM model

to each model's results individually. Both the Weather DNN and Multi-Temporal CNN-RNN are trained individually before training the Hybrid, which reduces the number of epochs needed to train the Hybrid, shown in Figure 6.9. The lowest achieved loss is 77.53 kg/daa, 6.6% lower than the Weather DNN and 5.3% lower than the Multi-Temporal CNN-RNNs individual results.

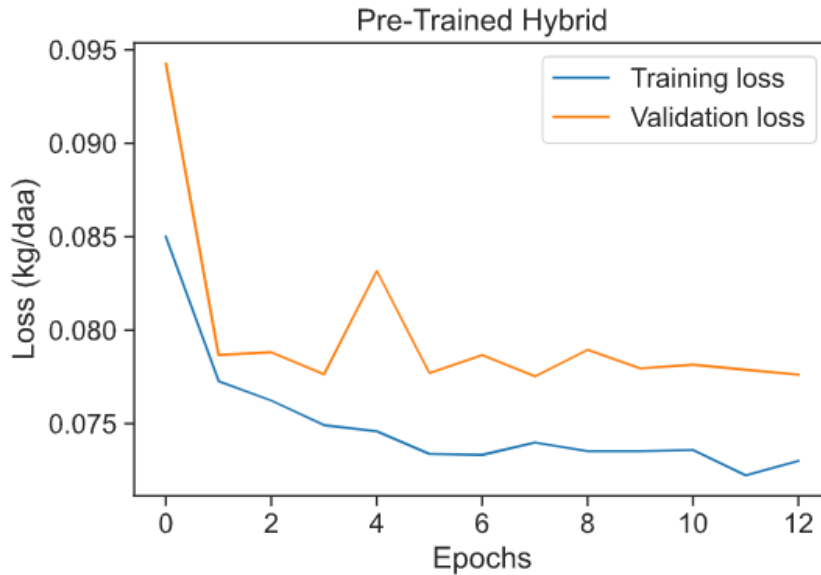


Figure 6.9: Training and validation loss for the Pre-Trained Hybrid model

The shorter training time for the Pre-Trained Hybrid may be explained by the pre-training of the two combined models. However, it may also indicate that the hybrid model is unable to find any valuable complex patterns that may exist between weather and satellite data, perhaps due to its architecture.

6.2.4 Hybrid 2: Hybrid CNN

The Hybrid CNN is the best performing model, and it manages so with fewer parameters and features than the Pre-Trained Hybrid, which contains additional features such as a farmers previous deliveries (historical yield). We attribute the Hybrid CNN's improvement to its more natural architecture that combines both weather and image data into a single sequence encoding. Compared to the Multi-Temporal CNN-RNN

which only looks at satellite images, the addition of weather data may allow the model to estimate plant growth where cloudy images would normally blind the model.

As the best performing model, we also test the model both with and without masking. As with the Single Image CNN experiment, we find that masking of the image provides significant improvements compared with no mask (see Figure 6.10 and Table 6.3), indicating that accurate field boundaries are highly useful for accurate crop yield predictions on a farm-scale. However, in contrast to the results from the single image experiment with masking, the Hybrid CNN model performs almost equally with both masks as a channel and masks applied. Although we have no conclusive answer as to why, one reason may be that the model starts overfitting earlier than the Single Image CNN and thus never reaches it's full potential.

| Mask type | Mean absolute error (kg/daa) |
|-----------------|------------------------------|
| No mask | 86.69 |
| Mask as channel | 76.57 |
| Mask applied | 76.27 |

Table 6.3: Hybrid CNN results with and without masks

As an additional analysis of the models accuracy, we present a Quantile-Quantile plot (Q-Q plot) in Figure 6.11 which compares the prediction output distribution with the real distribution. The Q-Q plot shows that the model learns a good approximation to the real distribution of crop yields from the validation set, although very low and very high yields seem to be more difficult to predict.

To visualise and better understand the models predictions we discretize predictions by grouping them into bins which are then plotted as a heatmap showing the cross-tabulation between prediction output and actual yield values. Figure 6.12 shows the discretized prediction outputs in equal width bins, which show that the model predicts well for the most common values. The figure may also explain why the model has difficulty predicting very high and very low yields, as there are much fewer samples of these to train on. A more balanced view of the prediction outputs is shown in Figure 6.13 where predictions are grouped into bins created from percentiles instead of a fixed width. The percentile bins show that predictions are centered roughly around the actual values on the diagonal, even for values in the top and bottom 10 %.

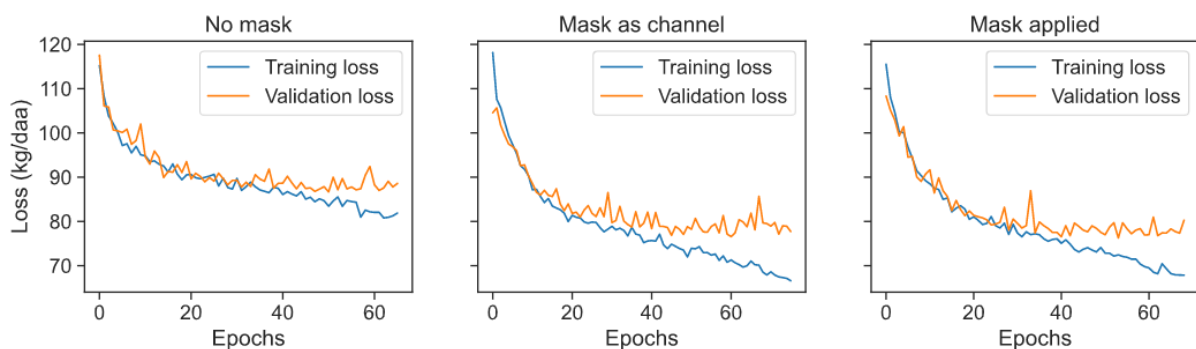


Figure 6.10: Training and validation loss for the Hybrid CNN

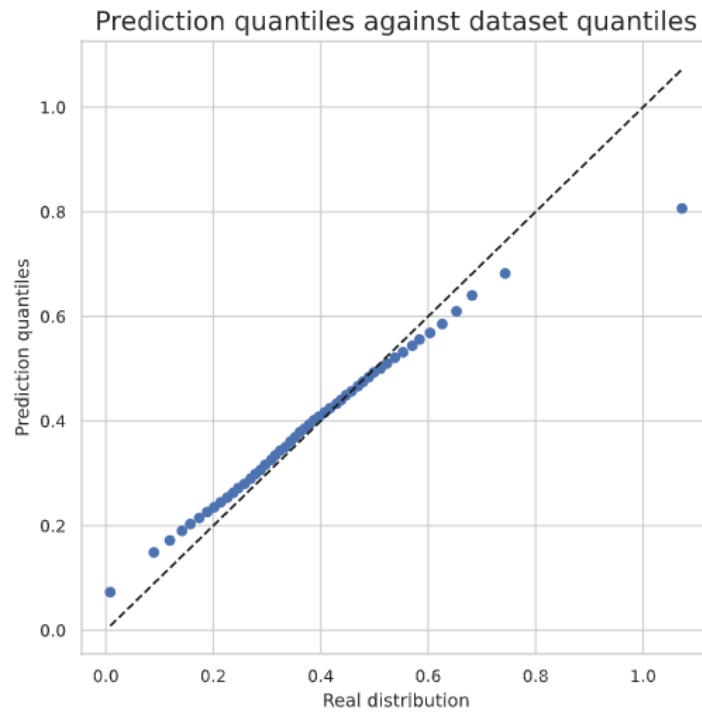


Figure 6.11: Hybrid CNN prediction quantiles versus real quantiles

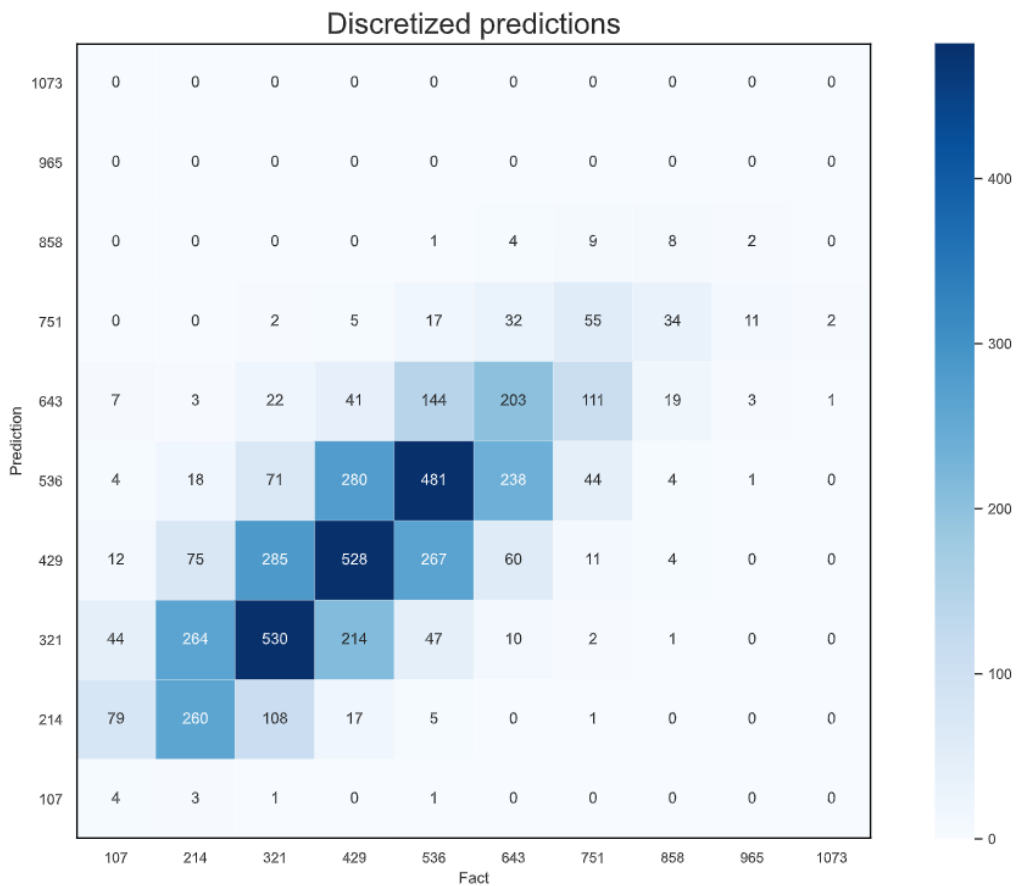


Figure 6.12: Discretized prediction output from the Hybrid CNN. Predictions are binned into ten bins. The axes' labels mark the upper bound of each bin.

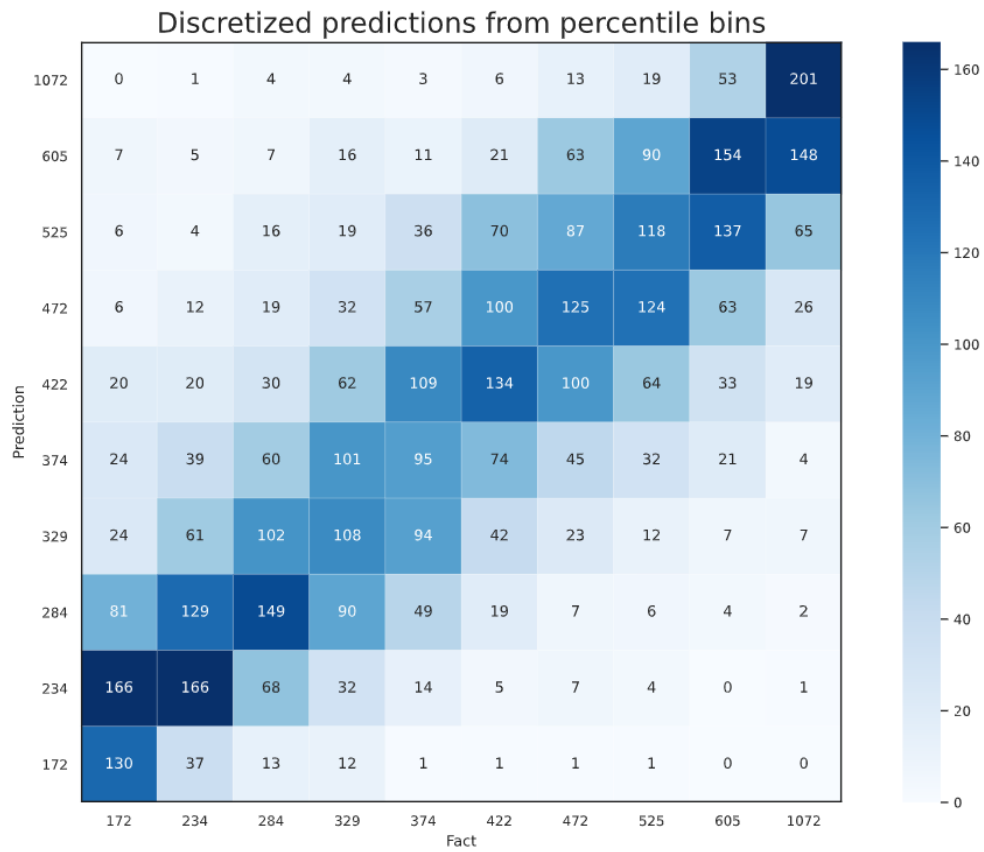


Figure 6.13: Discretized prediction output from the Hybrid CNN in percentile bins. Predictions are binned into ten percentile bins (10, 20, ..., 100) derived from the actual distribution. The axes' labels mark the upper bound of each percentile bin.

6.3 Early Predictions

In Hypothesis 4 we hypothesize that satellite and weather time series data can be used to predict crop yields before harvest. To test how much the early and middle periods of the growing season affect crop yield, we reuse the best performing model, the Hybrid CNN with masks applied, to predict crop yields without the full data time series. The single image experiments have already shown that satellite images from week 26 appears to contain the most relevant information out of all the weeks, implying that data after week 26 might not be as important to crop yield. By training the Hybrid CNN on shorter time series we can compare the accuracy of early predictions versus predictions on the whole time series. Table 6.4 show the mean absolute errors achieved when limiting the amount of data to 12 and 17 weeks of data, roughly equal to mid-May and late-June predictions respectively.

| Input | Description | Mean Absolute Error | Change |
|--------------|--------------------|----------------------------|---------------|
| Weeks 10-39 | Full season | 76.27 kg/daa | - |
| Weeks 10-26 | Late-June | 82.11 kg/daa | +7.66 % |
| Weeks 10-21 | Mid-May | 92.20 kg/daa | +20.89 % |

Table 6.4: The mean absolute error achieved at different times with early predictions. The error increases by 7.66 % for late-June predictions and 20.89 % for mid-May predictions compared to predictions made using the whole season.

As expected, the error increases when predicting earlier in the growing season. The late-June predictions, which includes all weeks up to and including week 26, has a moderate increase in error that is still lower than predictions made by other models on the full season. Figures 6.14 and 6.15 show the predictions in percentile bins for late-June and mid-May respectively. Mid-May predictions show a clear reduction in accuracy compared to both late-June and full season predictions (Figure 6.13), as the model struggles to differentiate between low and medium yields.

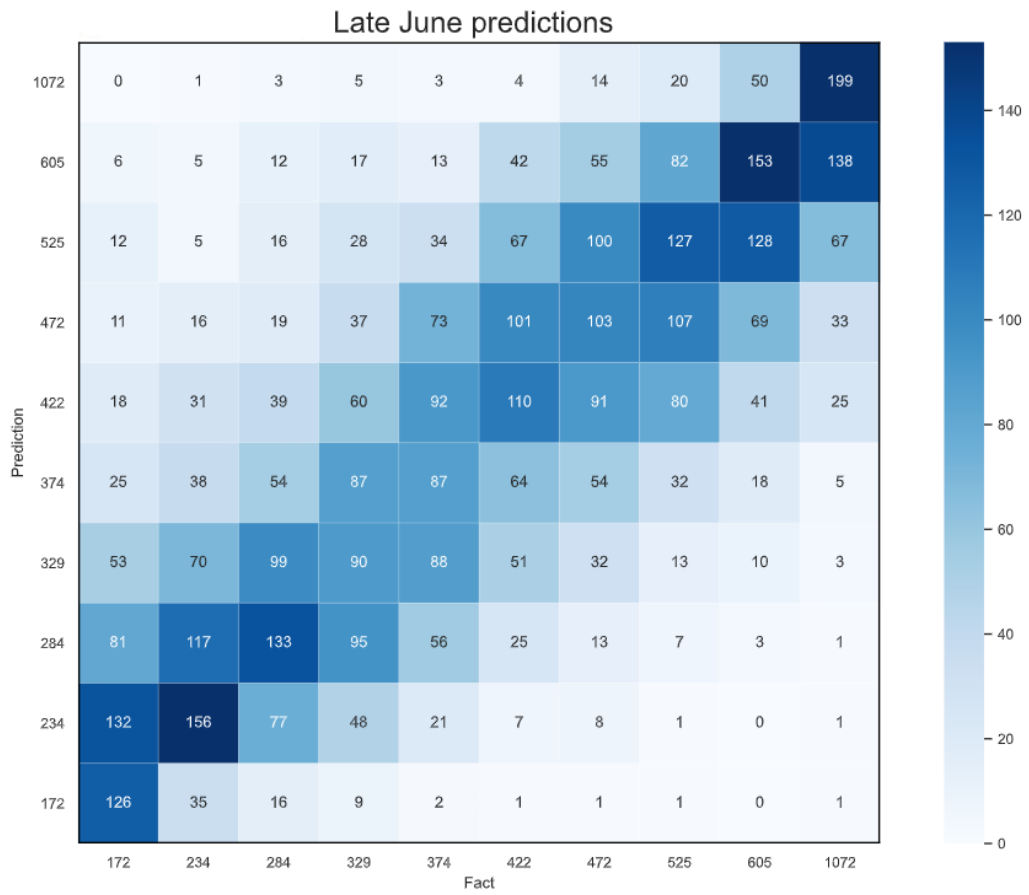


Figure 6.14: Early Predictions: Late-June

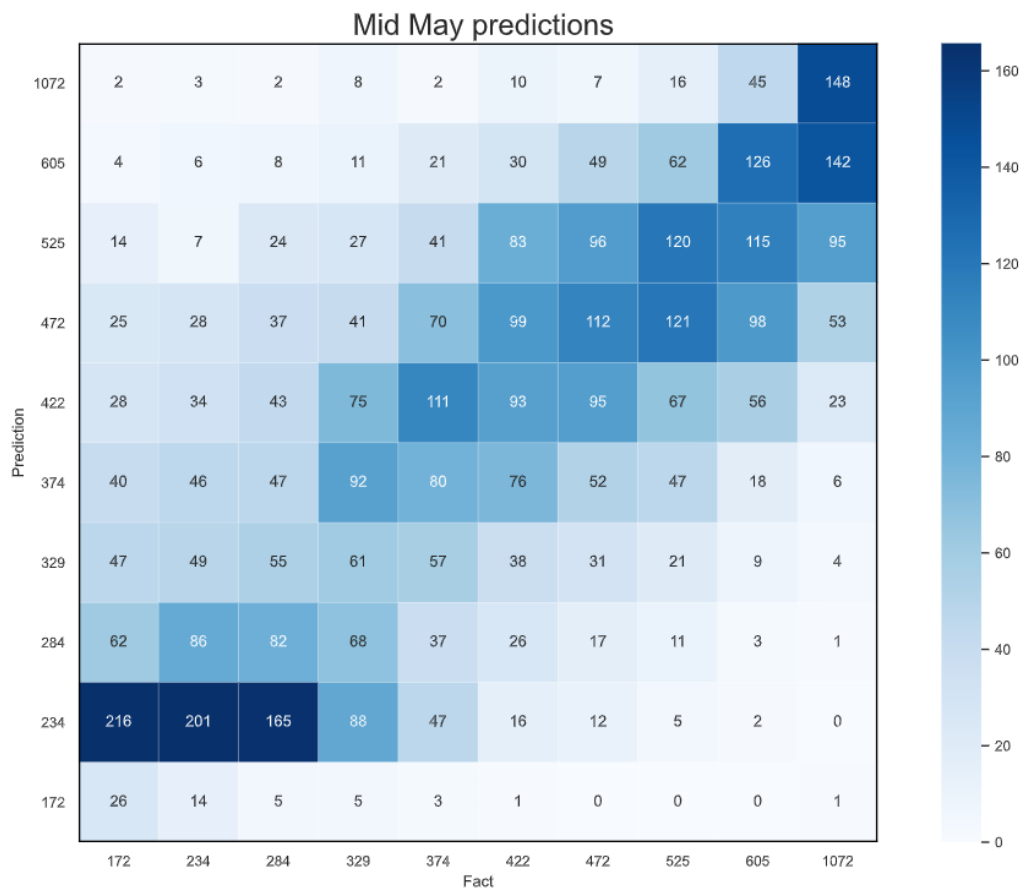


Figure 6.15: Early Predictions: Mid-May

6.4 Predictions as Regional Analytics

To the best of our knowledge, there is no prior work that predicts real world farm-scale crop yields on a large scale, making it difficult to compare our results with the current state-of-the-art in crop yield predictions. Sharma et al. [35] predict wheat yield at a tehsil scale, a small administrative unit in India, and achieve a RMSE of between 4.8 and 33 kg/daa when trained on different states in India. To compare with our results, we aggregate predictions made on a farm-scale up to the lowest administrative unit, the Norwegian commune, by assuming a mean farm-scale prediction per year for each commune. As some communes have very few samples, we take the 100 communes with the highest number of samples in our dataset to analyse¹. With this approach, our best performing model achieves a nation wide commune-scale crop yield prediction with MAE of 23.35 kg/daa and a RMSE of 30.81 kg/daa, which indicates that its accuracy is in the range of the predictions made by Sharma et al. on a tehsil scale in India². Figure 6.16 shows the relationship between predicted and actual commune-scale crop yield, made by aggregating farm-scale yields.

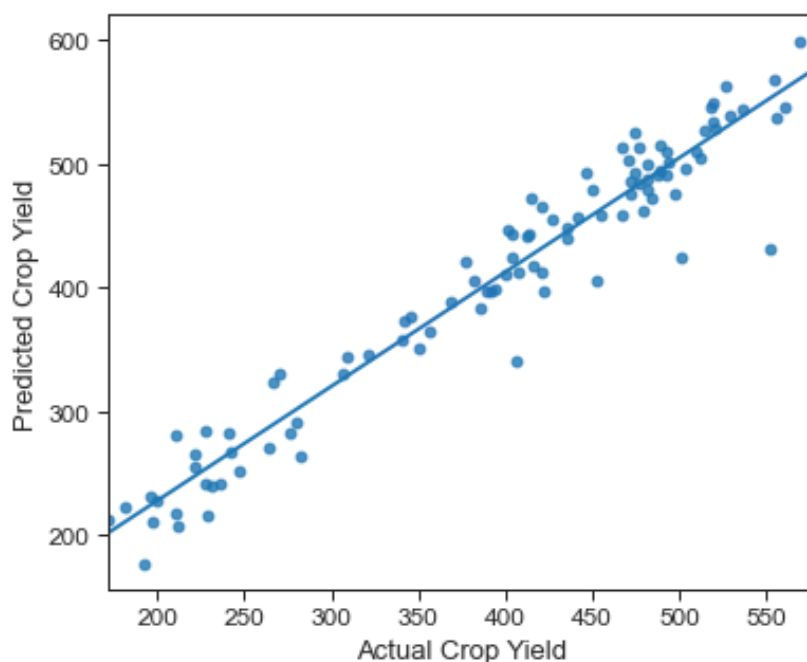


Figure 6.16: Relationship between actual and predicted crop yields on a commune-scale

¹The number of samples in the top 100 communes ranges from 32 to 152, with a mean of 60.

²As there are too many uncontrolled variables between these projects, this comparison only provides a rough idea of the performance level of two models that were not meant to predict the same thing.

6.5 Summary

Evaluating the initial experiments, the single image CNN is able to find a correlation between the multispectral satellite images and the yield of the farms. Further, the data augmentation methods of rotating, cropping, and adding noise to the images all seem to be effective, indicating that the model’s accuracy could improve given additional data. The application of image masks also provided positive results, showing that highlighting or keeping only the farms’ cultivated fields in the images enables the model to focus on the relevant portions of the image.

The best performing model is the Hybrid CNN, which utilizes both weather data and raw multispectral satellite images as its input, and achieves an improvement of 8 % over the baseline Weather DNN. Additionally, seemingly in accordance with the findings of You et al. [44] and Sharma et al. [35], using raw multispectral images outperforms the model using handcrafted vegetation indices. The Hybrid CNN is also capable of making early, in-season predictions, though the error increases when the amount of data decreases. In an effort to make the results of the proposed Hybrid CNN comparable to earlier works, we average the models per-farm predictions to predict on a per-commune basis. We see that the per-commune predictions of the Hybrid CNN are comparable to the latest state-of-the-art in crop yield predictions.

Chapter 7

Conclusions

In conclusion, this thesis explores the use of satellite data for crop yield prediction, reviewing both traditional and new methods of extracting relevant information from satellite imagery. A new dataset with real world per-farm samples, created by combining a multitude of data sources, enables the use of deep learning for farm-scale crop yield prediction. Multiple models are proposed and used to test different hypotheses and optimize prediction accuracy. The most accurate model is a deep convolutional, recurrent, and hybrid model that combines multispectral satellite images and weather data to predict crop yields. To the best of our knowledge, the model is a first of its kind in predicting farm-scale crop yields. In addition, we show that by aggregating farm-scale predictions to a commune scale, our model achieves comparable results to the current state-of-the-art on crop yield predictions.

All the hypotheses defined for this thesis are tested, and we restate them below along with a brief conclusion for each.

Hypothesis 1: *Satellite images of farms and their surroundings can be used to accurately predict farm-scale crop yields.*

Conclusion: Satellite images were used to predict farm-scale crop yields, and the results show that models using satellite images improves the prediction accuracy over the baseline model using weather data.

Hypothesis 2: *Accurate field boundaries along with satellite images increase crop yield accuracy significantly.*

Conclusion: We show that accurate field boundaries, represented using pixel masks along with satellite images, increases prediction accuracy significantly using the best performing model. The best performing model saw a 14% reduction of mean absolute error with the pixel masks, from 86.69 kg/daa without to 76.27 kg/daa with mask.

Hypothesis 3: *Prediction accuracy can be further increased by combining satellite images and meteorological data.*

Conclusion: Prediction accuracy was consistently better with models that incorporate both weather data and satellite images, suggesting that both contain some

information which the other does not.

Hypothesis 4: *It is possible to predict farm-scale crop yield earlier in the growing season with some reduced accuracy.*

Conclusion: By training the best hybrid model using data from earlier in the season, we show that late-June predictions can be done with a moderate increase in mean error (7.66 %) while mid May predictions are significantly less accurate with a almost 21 % error increase.

Chapter 8

Future Work

Given that this is a novel application of neural networks in a domain where data is limited and noisy, there are many untested methods and data sources that could improve prediction accuracy or achieve similar results more efficiently. This chapter briefly reviews some of our suggested directions for further research.

8.1 Improving Generalization

While our models show that accurate farm-scale crop yield predictions are possible with deep learning, the majority of models start to exhibit overfitting when training, even with the data augmenting methods used. This suggest that even higher accuracy might be possible given more data, or by using other known methods for reducing overfitting and increasing generalization such as batch normalization.

8.2 Remote Sensed Temperature

Land surface temperatures derived from satellite sensors have successfully been used in US county level predictions[17], and provide temperature values that should be closer to the actual temperature at the farm compared to interpolations between sensors that are typically many kilometers away from the farm. Such values could possibly replace temperature interpolation, or be used to improve weather interpolation further.

8.3 NIBIO Field Data

Apply field data gathered by NIBIO to the models. Adding features such as soil quality and water storage capacities could add meaningful information about crop yield potential in individual fields.

8.4 Additional Sources for Satellite Images

This thesis used Sentinel-2 as the source for the satellite images. It could be positive to introduce additional sources for satellite images to complement the Sentinel-2 dataset. Additional sources for satellite images could increase the frequency of the

satellite images throughout the growing season and give higher resolution multispectral images. Some specific satellites of interest could be Landsat 8, WorldView-3, and PlantScope.

Bibliography

- [1] Charu C Aggarwal. *Neural Networks and Deep Learning: A Textbook*. eng. Cham: Springer International Publishing AG, 2018. ISBN: 3319944622.
- [2] Mauritz Åssveen and Unni Abrahamsen. “Varmesum for sorter og arter av korn.” no. In: vol. 2/99, pp. 55–59. ISBN: 82-479-0109-9. URL: <https://www.nb.no/nbsok/nb/4462df6c3dc98eb25fb957008eeb2d57?lang=no#57>.
- [3] P. Basnyat et al. “Optimal time for remote sensing to relate to crop grain yield on the Canadian prairies.” English. In: *Canadian Journal of Plant Science* 84.1 (2004). Cited By :31, pp. 97–103.
- [4] Roberto Benedetti and Paolo Rossini. “On the use of NDVI profiles as a tool for agricultural statistics: The case study of wheat yield estimate and forecast in Emilia Romagna.” In: *Remote Sensing of Environment* 45.3 (1993), pp. 311–326. ISSN: 0034-4257. DOI: [https://doi.org/10.1016/0034-4257\(93\)90113-C](https://doi.org/10.1016/0034-4257(93)90113-C). URL: <https://www.sciencedirect.com/science/article/pii/003442579390113C>.
- [5] Yaping Cai et al. “Integrating satellite and climate data to predict wheat yield in Australia using machine learning approaches.” In: *Agricultural and Forest Meteorology* 274 (2019), pp. 144–159. ISSN: 0168-1923. DOI: <https://doi.org/10.1016/j.agrformet.2019.03.010>. URL: <https://www.sciencedirect.com/science/article/pii/S0168192319301224>.
- [6] Kyunghyun Cho et al. *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. 2014. arXiv: 1406.1078 [cs.CL].
- [7] Kunihiko Fukushima and Sei Miyake. “Neocognitron: A new algorithm for pattern recognition tolerant of deformations and shifts in position.” In: *Pattern Recognition* 15.6 (1982), pp. 455–469. ISSN: 0031-3203. DOI: [https://doi.org/10.1016/0031-3203\(82\)90024-3](https://doi.org/10.1016/0031-3203(82)90024-3). URL: <https://www.sciencedirect.com/science/article/pii/0031320382900243>.
- [8] Bo-cai Gao. “NDWI—A normalized difference water index for remote sensing of vegetation liquid water from space.” In: *Remote Sensing of Environment* 58.3 (1996), pp. 257–266. ISSN: 0034-4257. DOI: [https://doi.org/10.1016/S0034-4257\(96\)00067-3](https://doi.org/10.1016/S0034-4257(96)00067-3). URL: <https://www.sciencedirect.com/science/article/pii/S0034425796000673>.
- [9] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : concepts, tools, and techniques to build intelligent systems*. eng. Second Edition. Sebastopol, CA: O’Reilly, 2019. ISBN: 9781492032649.

- [10] Anatoly A. Gitelson. “Wide Dynamic Range Vegetation Index for Remote Quantification of Biophysical Characteristics of Vegetation.” In: *Journal of Plant Physiology* 161.2 (2004), pp. 165–173. ISSN: 0176-1617. DOI: <https://doi.org/10.1078/0176-1617-01176>. URL: <https://www.sciencedirect.com/science/article/pii/S0176161704705726>.
- [11] Ingvild Haugen et al. *Kornproduksjon i møte med klimaendringer, Et mer klimarobust landbruk i Vestfold og Telemark*. ISBN 978–82-336-0244-4 491. Nov. 2019.
- [12] Douglas M. Hawkins. “The Problem of Overfitting.” en. In: *Journal of Chemical Information and Computer Sciences* 44.1 (Jan. 2004), pp. 1–12. ISSN: 0095-2338. DOI: [10.1021/ci0342472](https://doi.org/10.1021/ci0342472). URL: <https://pubs.acs.org/doi/10.1021/ci0342472> (visited on 06/02/2021).
- [13] Sepp Hochreiter. “The Vanishing Gradient Problem During Learning Recurrent Neural Nets and Problem Solutions.” In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6 (Apr. 1998), pp. 107–116. DOI: [10.1142/S0218488598000094](https://doi.org/10.1142/S0218488598000094).
- [14] Sepp Hochreiter and Jürgen Schmidhuber. “Long Short-term Memory.” In: *Neural computation* 9 (Dec. 1997), pp. 1735–80. DOI: [10.1162/neco.1997.9.8.1735](https://doi.org/10.1162/neco.1997.9.8.1735).
- [15] Sepp Hochreiter et al. *Gradient Flow in Recurrent Nets: the Difficulty of Learning Long-Term Dependencies*. 2001.
- [16] Hao Jiang et al. “A deep learning approach to conflating heterogeneous geospatial data for corn yield estimation: A case study of the US Corn Belt at the county level.” eng. In: *Global change biology* 26.3 (2020), pp. 1754–1766. ISSN: 1354-1013.
- [17] David M. Johnson. “An assessment of pre- and within-season remotely sensed variables for forecasting corn and soybean yields in the United States.” In: *Remote Sensing of Environment* 141 (2014), pp. 116–128. ISSN: 0034-4257. DOI: <https://doi.org/10.1016/j.rse.2013.10.027>. URL: <https://www.sciencedirect.com/science/article/pii/S0034425713003957>.
- [18] Meetpal S Kukal and Suat Irmak. “Climate-Driven Crop Yield and Yield Variability and Climate Change Impacts on the U.S. Great Plains Agricultural Production.” eng. In: *Scientific reports* 8.1 (2018), pp. 3450–18. ISSN: 2045-2322.
- [19] *Landsat 1 | Landsat Science*. URL: <https://landsat.gsfc.nasa.gov/landsat-1-3/landsat-1> (visited on 04/06/2021).
- [20] Y. Lecun et al. “Gradient-based learning applied to document recognition.” In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791).
- [21] David B. Lobell et al. “A scalable satellite-based crop yield mapper.” In: *Remote Sensing of Environment* 164 (2015), pp. 324–333. ISSN: 0034-4257. DOI: <https://doi.org/10.1016/j.rse.2015.04.021>. URL: <https://www.sciencedirect.com/science/article/pii/S0034425715001637>.
- [22] R. Macdonald, Forrest Hall, and R. Erb. “The use of Landsat data in a Large Area Crop Inventory Experiment (LACIE).” In: *[No source information available]* (Feb. 1975). Phase I.
- [23] R. B. MacDonald. “The Large Area Crop Inventory Experiment.” In: Phase I and Phase II. Sioux Falls, South Dakota: Falls Church, Va.: The Society, 1977, Oct. 1976.

- [24] *MSI Instrument - Sentinel-2 MSI Technical Guide - Sentinel Online - Sentinel*. URL: <https://sentinel.esa.int/web/sentinel/technical-guides/sentinel-2-msi/msi-instrument> (visited on 04/15/2021).
- [25] P. Nejedlik, R. Oger, and R. Sigvald. "The phenology of crops and the development of pests and diseases." In: 1999.
- [26] Ingvild Nystuen and Siri Svengaard-Stokke. *Jordsmonn - Organisk materiale - Kartkatalogen*. URL: <https://kartkatalog.geonorge.no/metadata/jordsmonn-organisk-materiale/6898f450-01ea-4b1c-b284-194308de1445> (visited on 05/14/2021).
- [27] Nathalie Pettorelli. *The Normalized Difference Vegetation Index*. eng. Oxford: Oxford University Press, 2013. ISBN: 0199693161.
- [28] *Reflectance Spectra*. URL: https://www.usna.edu/Users/oceano/pguth/md_help/html/ref_spectra.htm (visited on 05/21/2021).
- [29] F. Rosenblatt. "The perceptron: A probabilistic model for information storage and organization in the brain." In: *Psychological Review* 65.6 (1958). Place: US Publisher: American Psychological Association, pp. 386–408. ISSN: 1939-1471(Electronic),0033-295X(Print). doi: 10.1037/h0042519.
- [30] J. W. Rouse Jr. et al. "Monitoring vegetation systems in the Great Plains with ERTS." In: *Goddard Space Flight Center 3d ERTS-1 Symp*. Vol. 1. NASA, Jan. 1974. URL: <https://ntrs.nasa.gov/citations/19740022614> (visited on 04/06/2021).
- [31] Vasit Sagan et al. "Field-scale crop yield prediction using multi-temporal WorldView-3 and PlanetScope satellite data and deep learning." eng. In: *ISPRS journal of photogrammetry and remote sensing* 174 (2021), pp. 265–281. ISSN: 0924-2716.
- [32] J.R. Schott. *Remote Sensing: The Image Chain Approach*. Oxford University Press, 2007. ISBN: 9780199724390. URL: <https://books.google.no/books?id=uoXvgw0zAkQC>.
- [33] R.A. Schowengerdt. *Remote Sensing: Models and Methods for Image Processing*. Elsevier Science, 2006. ISBN: 9780080480589. URL: <https://books.google.no/books?id=KQXNaDHOX-IC>.
- [34] Y. Shao, J. Ren, and J. B. Campbell. "Multitemporal Remote Sensing Data Analysis for Agricultural Application." In: 2018.
- [35] Sagarika Sharma, Sujit Rai, and Narayanan C. Krishnan. "Wheat Crop Yield Prediction Using Deep LSTM Model." In: *CoRR* abs/2011.01498 (2020). arXiv: 2011.01498. URL: <https://arxiv.org/abs/2011.01498>.
- [36] Benjamin Sjølander, Erik Sandø, and Engen Martin. *Neural Network for Grain Yield Predictions in Norwegian Agriculture*. 2020. URL: https://github.com/putetrek/kornmo/blob/244bc953db2cf8be64ca5fd8396d859ce54f93f1/documents/NN_for_Yield_Prediction.pdf.
- [37] Oregon State University. *Environmental Factors Affecting Plant Growth*. <https://extension.oregonstate.edu/gardening/techniques/environmental-factors-affecting-plant-growth>. Accessed: 2020-12-07. URL: <https://extension.oregonstate.edu/gardening/techniques/environmental-factors-affecting-plant-growth> (visited on 12/07/2020).

- [38] Thomas van Klompenburg, Ayalew Kassahun, and Cagatay Catal. “Crop yield prediction using machine learning: A systematic literature review.” In: *Computers and Electronics in Agriculture* 177 (2020), p. 105709. ISSN: 0168-1699. DOI: <https://doi.org/10.1016/j.compag.2020.105709>. URL: <https://www.sciencedirect.com/science/article/pii/S0168169920302301>.
- [39] R. Venkatesan and B. Li. *Convolutional Neural Networks in Visual Computing: A Concise Guide*. Data-enabled engineering. CRC Press, 2018. ISBN: 9781138747951. URL: <https://books.google.no/books?id=Y2xSAQAACAAJ>.
- [40] Pascal Vincent et al. “Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion.” In: *Journal of machine learning research* 11.12 (2010).
- [41] Diana I. Walker, Birgit Olesen, and Ronald C. Phillips. “Chapter 3 - Reproduction and phenology in seagrasses.” In: *Global Seagrass Research Methods*. Ed. by Frederick T. Short and Robert G. Coles. Amsterdam: Elsevier Science, 2001, pp. 59–78. ISBN: 978-0-444-50891-1. DOI: <https://doi.org/10.1016/B978-044450891-1/50004-9>. URL: <https://www.sciencedirect.com/science/article/pii/B9780444508911500049>.
- [42] Anna X. Wang et al. “Deep Transfer Learning for Crop Yield Prediction with Remote Sensing Data.” In: *Proceedings of the 1st ACM SIGCAS Conference on Computing and Sustainable Societies*. COMPASS '18. Menlo Park and San Jose, CA, USA: Association for Computing Machinery, 2018. ISBN: 9781450358163. DOI: [10.1145/3209811.3212707](https://doi.org/10.1145/3209811.3212707). URL: <https://doi.org/10.1145/3209811.3212707>.
- [43] Craig L Wiegand et al. “Development of Agrometeorological Crop Model Inputs from Remotely Sensed Information.” eng. In: *IEEE transactions on geoscience and remote sensing* GE-24.1 (1986), pp. 90–98. ISSN: 0196-2892.
- [44] Jiaxuan You et al. “Deep Gaussian Process for Crop Yield Prediction Based on Remote Sensing Data.” In: *Proceedings of the AAAI Conference on Artificial Intelligence* 31.1 (Feb. 2017). URL: <https://ojs.aaai.org/index.php/AAAI/article/view/11172>.