Student thesis series INES nr 668

Wild boar damage mapping in agricultural grass and wheatlands using Unmanned Aerial Vehicle (UAV) data

Yrsa Ine Maria Kleijkers

2024

Department of

Physical Geography and Ecosystem Science

Lund University

Sölvegatan 12

Yrsa Ine Maria Kleijkers (2024).
Wild boar damage mapping in agricultural grass and wheatlands using Unmanned Aerial Vehicle (UAV) data
Master degree thesis, 30 credits in *GIS and Remote Sensing*
Department of Physical Geography and Ecosystem Science, Lund University

Level: Master of Science (MSc)

Course duration: *January* 2024 until *June* 2024

Wild boar damage mapping in agricultural grass and wheatlands using Unmanned Aerial Vehicle (UAV) data

---

Yrsa Ine Maria Kleijkers

Master thesis, 30 credits, in *Master of Science (MSc)*

Florent Rumiano
Sveriges Lantbruksuniversitet SLU Grimsö forskningsstation

Petter Kjellander
Sveriges Lantbruksuniversitet SLU Grimsö forskningsstation

Ulrik Mårtensson
Lund University

Exam committee:
Per-Ola Olsson, Lund University
Babak Mohammadi, Lund University

Abstract

The growth of the wild boar (*Sus Scrofa*) population in Sweden results in increasing wild boar damage on agricultural grounds, thereby influencing the livelihoods of farmers. Finding mitigation strategies that are positive for both wildlife and human society is challenging. It is therefore essential to enable the creation and investigation of precise and differentiated wild boar management strategies. This necessitates the development of semi-automatized and trusted methods that quantitatively and spatially assess wild boar damages on agricultural lands at the landscape scale (i.e., cm-level). Current methods cannot answer this need as they consist of manual field surveys that are time-consuming, subjective, and have a too-coarse scale. Unmanned Aerial Vehicle (UAV) can be a solution to systematically assess damages in agricultural fields on this landscape scale, as it provides data with high temporal and spatial scale, flexibility, and affordability. This study developed three methods that utilized UAV data to automatically map (wild boar) damages in agricultural grass and wheatlands in Boo, Hjortkvarn Municipality, Sweden. Two methods, pixel-based and object-based classifications, focused on performing an image classification on 2-dimensional (2D) multispectral UAV data by applying the machine learning algorithms Random Forest (RF) and Support Vector Machine (SVM). The object and pixel-based RF and SVM classification retrieved overall accuracies of 85% and above for the wheatlands and 91% and above for the grassland, whereby the best overall performance was achieved by the object-based SVM method for both wheat and grassland. The two classification methods created damage maps with similar damage locations but the pixel-based classification mapped the damage extents smaller compared to the object-based classification. The third method utilized the 3-dimensional (3D) UAV photogrammetry-derived point clouds of the wheatlands. This method extracted the normalized height values of the point cloud and applied a height threshold to create the damage mapping. The method was not able to capture the damage mapping in wheatlands due to the too-low density point clouds but has the potential to aid the 2D classifications by providing the extracted normalized height values as additional information for the machine learning classifiers.

Acknowledgment

I want to thank Dr. Florent Rumiano, Prof. Petter Kjellander and Arvid Norström from SLU (The Swedish University of Agricultural Sciences) Grimsö Wildlife Research Station for their support in this Master's thesis. Dr. Florent Rumiano supervised all aspects of this master's thesis. The methodology used in this thesis is an adaptation of his developed methodology and is part of a project from Prof. Petter Kjellander at the SLU Grimsö Wildlife Research Station, Department of Ecology, funded by Naturvårdsverket (EPA). This thesis's developed scripts in the programming language R, are my transcription of the methodology based on a collaboration between Dr. Florent Rumiano and me. The data used in this study was collected by Arvid Norström and Dr. Florent Rumiano. I also want to thank Dr. Per-Ola Ollsen and Ulrik Mårtensson from Lund University for providing support and feedback.

Table of Contents

Abbreviations

2D – 2-Dimensional

3D – 3-Dimensional

B – Blue (wavelength band)

CNN – Convolutional Neural Network

DEM – Digital Elevation model

DSM – Digital Surface Model

G – Green (wavelength band)

GLCM – Gray-Level Co-Occurrence Matrix

GNSS – Global Navigation Satellite System

GSD – Ground Sampling Distance

HWC – Human-wildlife Conflict

IDW – Inverse Distance Weighting

IUCN – International Union for Conservation of Nature

JM – Jeffries-Matusita

LIDAR – Light Detection and Ranging

LULC – Land Use and Land Cover

ML – Machine Learning

NDVI – Normalized Difference Vegetation Index

NIR – Near-infrared (wavelength band)

R – Red (wavelength band)

RE – Red Edge (wavelength band)

RF – Random Forest

RGB – Red, Green, Blue (wavelength band)

SCB – Statistics Sweden

SVM – Support Vector Machine

UAV – Unmanned Aerial Vehicle

1. Introduction

In today's world, human societies face many challenges to coexist with wildlife. Amongst these challenges is the increasing human-wildlife conflict (HWC), resulting from enlarging wildlife habitat loss due to factors such as human exploitation of limited natural resources (e.g. expansion of agriculture and urbanization), population growth and climate change (Ravenelle & Nyhus, 2017; Torres et al., 2018; International Union for Conservation of Nature [IUCN], 2022). HWC entails, in one respect, the direct conflict between wildlife and human societies (e.g. crop damages, livestock kills, and road accidents). In Sweden, a direct HWC is caused by the growth of its wild boar (*Sus Scrofa*) population, resulting in increasing damages on agricultural grounds, thereby influencing the livelihoods of farmers. The annual growth rate of the wild boar in Sweden was 50% between 2000-2010 (Gren et al., 2019) and is currently estimated to be around 300,000 individuals (Swedish Association for Hunting and Wildlife Management). Wild boar cause damage to almost all agricultural crops in Sweden with the highest losses for wheat (Edman & Ländell, 2021). The losses of wheat yield in 2020 totaled 84,400 tonnes (i.e., 1.4% of the total wheat harvest), which is double the losses caused by wild boar in 2014. The negative consequences of wild boar rooting behavior in agricultural lands were estimated to be around 17% of Sweden's average net farm income in 2015 (Gren et al., 2019). These negative consequences are due to cost from crop damages (e.g., yield loss) and additional costs (Rao et al., 2002), such as costs related to machinery damage, reduced soil quality, and protection (e.g., hunting and field fencing).

Finding mitigation strategies for these direct conflicts between wildlife and human societies is a challenging task. For example, the mitigation strategy called the Hunting Act (Jaktlagen), implemented in 1966 in Sweden, had the desired outcome of an increase in the wolf population, but brought negative consequences for some, resulting in a divide between 'pro-wolf' and 'anti-wolf' people within the Swedish society (Lin et al., 2021). This shows that within the HWC, conflict can also arise between groups of people (IUCN) such as environmental policymakers, urban planners, farmers, and indigenous people (e.g. Sami people) (Sjoegren and Matsuda, 2016). These conflicts are a result of the differences in each group's perceptions, which are framed by past experiences, beliefs, and values (e.g. what is the place of the human in nature) (Peterson et al., 2010). Finding the right mitigation techniques is complex due to this social influence. Other challenges of mitigation strategies are related to financial constraints and the unavailability of resources in specific regions. So are technological mitigations (Fall and Jackson, 2002; Dickman, 2010) such as the use of electronic collars (Rossler et al., 2012), often non-ideal because financial limitations restrict their large-scale deployment.

To comprehend an HWC, such as the negative impact of wild boar activities on Sweden's agricultural sector, studies (Rostro-García et al., 2016; Jin et al., 2021) have researched the potential of risk models. Jin et al. (2021) showed that a risk model could aid precise and differentiated management strategies for wild boar management in Hunchun, China. These risk models use historical damage locations to enable the assessment of HWC risks and provide possibilities to investigate different mitigation strategies (Miller, 2015). Studies on risk models show that with the availability of systematic time-series data on historical damage locations (e.g., wild boar damages on agricultural grounds), wildlife managers and policymakers (e.g., The Swedish Environmental Protection Agency) can have the knowledge and the ability to discover temporal and spatial patterns of HWC. This is crucial information for the development of mitigation interventions that are beneficial for all parties (i.e. humans and wildlife) (Ding et al., 2023).

In this environmental and economic context, there is a need to develop semi-automatized and trusted methods, that quantitatively and spatially assess wild boar damages on agricultural fields at the landscape scale (i.e., Very-high spatial resolution, cm-level). By working with information on a high scale (i.e. cm-level), precise management strategies can be created that have a lower chance of negatively impacting parts of the wildlife and human society. The creation of semi-automatized methods is important, as it will increase the usability of the method by a broad range of potential users, as specific technical knowledge will be less required when utilizing the method.

To my knowledge, no systematic data collection of damage locations is done on agricultural lands in Sweden. Current methods consist of manual field surveys executed by authorized people (e.g. Swedish Agency for Agriculture (Jordbruksverket)). They are limited by being time-consuming, subjective, error-prone, and may cause additional crop damage during surveying (Dabrowska-Zielinska et al., 2009). Also, the spatial scale of the data collection is often too coarse to be useful in applications such as HWC risk modeling. This spatial scale issue is visible in a large-scale data collection report that was published by Statistics Sweden (SCB) in collaboration with the Swedish Agency of Agriculture in 2020 (SCB and Jordbruksverket, 2020). This report provided statistics on wildlife damages on agricultural lands in Sweden on county and production area scale (i.e. total counts and percentages of destructed area). Although this data can provide insights into wildlife damages and its implications on a large scale, for the current need for precise and differentiated mitigation management, it is crucial to have landscape scale (i.e. separate agricultural fields) data with very-high spatial resolution.

Over the past four decades, remote sensing data such as Unmanned Aerial Vehicle (UAV) has been widely used by scientists for agriculture applications (Mulla, 2013). For instance, studies conducted by Wang et al. (2018) and Ge et al. (2019) utilized UAV data for biomass estimation, Feng et al. (2015) for vegetation mapping, and Su (2017) for water quality monitoring. A valuable characteristic of UAVs is their ability to collect data in an automated and repeatable manner (i.e. via setup flight plans) with very-high spatial (i.e. <1m) and temporal resolution scales. Also, the high flexibility that exists within the UAV data collection enables the collection of a wide range of data. This flexibility is reflected in the instrumentation adaptability of the UAV (e.g., angle of observation, flight route, flight height, flight speed, and sensor types), flying time decisions (e.g., controlling the weather conditions during the flight), and the options for live data transmissions (Pádua et al., 2017).

Satellite data, another form of remote sensing data, has also been proven valuable in agricultural applications (Skakun et al., 2016; Nguyen et al., 2020). It can cover large areas and can collect data without the need for an operator to be physically present at the study site. However, the fixed time acquisition (Müllerová et al., 2017) and environmental factors, such as cloud coverage, affect the usability of this data in space and time. The spatial resolution of freely available satellite data, such as the Sentinel-2, is lower (e.g., several meters) than the UAV data (e.g., cm-level). The kind of information that can be extracted from the data is therefore different between the satellite and UAV (Woodcock and Strahler, 1987; Rocchini, 2007). Commercial satellite images can have better spatial resolutions, however, prices of purchasing satellite images can become very high, this depends on factors such as the amount of preprocessing (e.g., atmospheric corrections), order size, spatial resolution and availability (Vailshery, 2022). For example, an order of one square kilometer of a WorldView-3 with an 8-band Multispectral and a spatial resolution of 1.24 meters costed around 3200 dollars in July 2023 (Hubing, 2023). Compared to these prices for separate satellite images, the

purchase of a UAV device is more affordable, for example, one unit of the UAV DJI Mavic 3 Multispectral (DJI, n.d.) costs around 53,000 SEK in March 2024.

The characteristics of UAV data (i.e. high temporal and spatial scale, high flexibility, and affordability) and the limitations of satellite data (i.e. lower temporal and spatial scale, low flexibility, and high costs) privilege the use of UAV data over satellite remote sensing for the application of mapping (wild boar) damages in agricultural fields on a landscape scale (i.e., cm-level).

2. Aims and Objectives

This master's thesis aims to contribute to the development of methods to map wild boar damages on very-high spatial scale resolution (i.e. 5 cm-level) in agricultural wheatlands (i.e. vertical growing crop) and grasslands (i.e. non-vertical growing crop) in Boo, Hjortkvarn Municipality, Sweden (Figure 1). Three different image classification methods utilizing UAV data have been developed, analyzed, and compared in performance. The first two methods, object-based classification and pixel-based classification, focused on two-dimensional (2D) UAV data and utilized the machine learning algorithms Random Forest (RF) and Support Vector Machine (SVM) (theoretical explanations in subsection 4.3.2.2). Their performances on (wild boar) damage mapping in agricultural wheat and grasslands were studied and compared. By comparing the two 2D methods, it could be investigated, whether the pixel-based classification (i.e. a computationally less demanding approach) could potentially be used as an alternative to the more computationally demanding object-based classification approach (more information subsection 4.2). The third method was developed for damage mapping in wheatlands and focused on three-dimensional (3D) UAV data (i.e. point cloud). The approach was based on the height differences between locations of damage and wheat that could be classified using a height threshold (i.e. point cloud height threshold classification). Studying the performance of the 3D-based approach gave insight into whether utilizing 3D UAV data could add benefits to the (wild boar) damage mapping in agricultural wheatlands.

This thesis focused on the development of objective, accurate, generalizable, open-source, and automated methods that utilize 2D-UAV and 3D-UAV data to map damages in agricultural wheat and grasslands within RStudio 2023.12.1 + 402 ("Ocean Storm" Release).

The following aim and research questions were addressed in this master's thesis.

Research aim:

> Automatically map damages (e.g. wild boar, environmental, machinery) on very-high spatial scale resolution (i.e. 5 cm) in agricultural wheat and grasslands in Sweden, using image classification techniques on multispectral 2D and 3D UAV data.

Research questions:

1. What are the performances of **object-based and pixel-based image classifications** on 2D multispectral UAV data using the machine learning classifiers RF and SVM to map damages in agricultural wheatlands and grasslands? Do they provide similar damage mappings?

2. What is the performance of a 3D-based UAV data approach, **point cloud height threshold classification**, on mapping damages in agricultural wheatlands? Does the method add any benefits for mapping damages in agricultural wheatlands?

Due to the high spatial scale and multispectral sensor data available by the utilization of UAV data captured from agricultural wheat and grasslands, and by the exploration of the three different damage mapping techniques (i.e. object-based, pixel-based, and point cloud height threshold classification), it is expected that a method can be created that automatically maps damages in agricultural grass and wheatlands in Boo, Hjortkvarn Municipality, Sweden.

For clarification, this study focused on mapping all damages present in the agricultural grass and wheatlands. Although the fields contained mostly damage originating from wild boar activities, damage originating from other sources such as machinery tracks, drought, and other wildlife, could have also been present in the agricultural fields. Secondly, no investigation has been done on whether the created methods for mapping damages in wheat and grassland can be used for the mapping of damages in other agricultural lands (e.g. sugar beet and potatoes). Thirdly, due to time and computational constraints, the scope of the study was set to the mentioned study area of Boo, Hjortkvarn Municipality. Even though there is a high potential that the developed methods will also work on agricultural wheat and grasslands in other regions within and outside of Sweden, this has not been tested for and can therefore not be presumed. Lastly, it should be clear that no discrimination could be made between mapped damages regarding the exact time of occurrence (e.g., is this, this year's damage or is this damage due to previous year's wild boar destructions), this as previous records from other years of damage locations were unavailable.

The upcoming sections will start with the context and background that provides an outline of studies that used UAV data to map wild boar damages in agricultural lands. Afterward, the methods (section 4) used in this study will be explained in detail, these entail the explanations of the study area and data collection (subsection 4.1), general methodology (subsection 4.2), object- and pixel-based classifications (i.e. 2D-methods, subsection 4.3), point cloud height threshold classification (3D-method, subsection 4.4) and the validation field data evaluation (subsection 4.5), after which the results (section 5) are presented. The results are then discussed (section 6) and lastly, a conclusion (section 7) is provided.

3. Context and Background

Processing of raw UAV images can provide both 2D UAV data in the form of ortho-mosaics (i.e. stitched together UAV images into one image) and 3D point cloud data (i.e. points in a 3D space with a location and height information). For both these 2D and 3D UAV data, methods have been studied to map damages in agricultural lands. This section will start with studies based on 3D-based methods.

3.1 Damage Mapping with 3-Dimensional Unmanned Aerial Vehicle Data

Studies utilizing 3D UAV data used UAV photogrammetric (i.e. based on multi-angle 2D UAV images, more information subsection 4.1) and Light Detection and Radar (LIDAR) (i.e. based on active sensor pulses, more information subsection 4.1) point clouds to map wild boar damages in agricultural lands with vertical growing crops (i.e. wheat and corn fields). Garcia Millan et al. (2020) studied the mapping of wheat and corn damages in sites with different climate and soil conditions in Europe and America. They utilized methods such as

slope detection (Burrough et al., 2015) and variance analysis (Anys et al., 1994) on the digital surface models (DSM) created from LIDAR point clouds. By detecting lowering changes in topographic elevations between neighboring pixels they were able to map severely damaged areas with an accuracy of 90%. Similar studies using methods based on crop and damage height differences used photogrammetry-derived point clouds. Kuželka and Surový (2018) performed a height analysis method for wheat loss estimation in the Czech Republic. Their method used kernels to detect height differences between wheat and damage in normalized height models (i.e. height above the surface) that were derived from the point clouds. Their method retrieved high accuracies (>90%), but had the disadvantages that prior knowledge of the sizes of the damages was required and that the method assumed similar crop heights throughout the field, an assumption that cannot always met. Michez et al. (2016) focused on wild boar damage mapping in cornfields in Belgium. They performed a subtraction of a LIDAR digital terrain model from their photogrammetry point cloud-derived DSM to obtain normalized height values. The normalized height differences between damage and corn were detected and they found that their model was underestimating the damaged areas with a maximum underestimation of almost 10%.

Even though these 3D UAV methods are promising for wild boar damage mapping in agricultural grounds, the 3D methods have their limitations. So are these methods relying on the height difference between damaged areas and non-damaged crops, making it mostly applicable to crops that grow in height (e.g. vertical) and not for low-height crops such as grass. Next to this, as advances for UAV platforms resulted in the availability of multispectral sensors (e.g. Red, Green, Blue and Near-infrared, thermal), a lot of different data is available for the mapping of wild boar damages (Eskandari et al.,2020). When only using the 3D data, this information is neglected. In addition, point clouds derived from 3D UAV data require larger storage space (e.g. multiple gigabytes per agricultural field of several hectares) compared to 2D UAV data.

3.2 Damage Mapping with 2-Dimensional Unmanned Aerial Vehicle Data

The extensive volume of available 2D UAV data per field has increased owing to its high spatial resolution and multispectral capabilities. Due to these extensive volumes of data, machine learning (ML) has been made possible and found to be a fast and reliable method for solving linear and non-linear problems for agricultural applications (Eskandari et al., 2020). The meta-analysis done by Eskandari et al. (2020) and the review of Belgiu and Dragut (2016) found that studies using UAV data for agricultural applications extensively apply the machine learning algorithms RF and SVM (for theoretical explanations refer to subsection 4.3.2.2.). Studies by Stas et al. (2016) and Heremans et al. (2015) used a SVM and RF respectively for wheat yield predictions where they found that small training samples and several spectral parameters could train SVMs and RFs to accurately predict the yields. A study highly related to wild boar damage mapping was recently done by Kumar et al. (2024). They focused on mapping rice damages due to lodging and trained an RF with several combined features such as spectral and textural information. They obtained an accuracy of over 95%. A study that used the classifier RF to assess wild boar damages in cornfields and grasslands was done in northern Belgium by Rutten et al. (2018). They made use of Geographic Object-Based Image Analysis from the eCognition® Developer software (Trimble Inc., Westminster, CO, USA), which transforms the input UAV data into objects of agglomerated pixels. The objects are groups of neighboring pixels that contain similar spectral values (Addink et al., 2010 and Blaschke, 2010). By using these objects as input data (i.e. object-based classification), additional information such as texture, shape and size could be provided to the RF classifier (Blaschke et al., 2014). They obtained an overall accuracy of

85% for the cornfields and 94% for the grasslands. Another study on wild boar damage mapping in cornfields was done by Friesenhahn et al. (2023) in Texas, USA. They ran a convolutional neural network (CNN) algorithm (more information subsection 6.7) available in ArcPro 2.7 (ESRI 2022). The input data to the CNN consisted of information extracted from a Red, Green, Blue (RGB) ortho-mosaic and a raster containing height values. They obtained an overall accuracy of over 80%. A SVM algorithm for mapping wild boar damages in cornfields in Mississippi, USA was applied by Samiappan et al. (2018). They used as well objects of pixels (i.e. object-based classification) to have additional texture information for the input of the SVM. Their retrieved classification accuracies were between 65% and 75%, where the classification underestimated the damages. Lastly, a study done by Fischer et al. (2019) focused on wild boar damage mapping in cornfields in Missouri, USA using a ML method that consisted as well of an object-based classification. To map the damages, a semi-automated and object-based software application called Feature Analyst (Overwatch Systems Ltd, 2010) was used. They found overall accuracies that ranged between 74% and 98%.

4. Methodology

In this section, the methods of analysis are presented together with the reasoning for choosing the specific methods. Keep in mind, however not written down explicitly in the upcoming sections, that all these methods are applied on both the grasslands and wheatlands separately.

4.1 Study Area and Data Collection

Given the time and computational constraints that were present in this master's thesis, the study area was narrowed to focus on agricultural wheat and grasslands in Boo, Hjortkvarn Municipality, southeast of Örebro County, Sweden (see Figure 1). This area is concerned with wild boar damages and the farmers in this area are in close contact due to other research-related studies. Boo can be characterized as a farmer's estate with agricultural lands that are in proximity of numerous forest patches and with a low human population density (i.e. 2.7 residents per hectare in 2020 (SCB)). Next to farming activities, hunting activities take place due to the high wildlife activity in this area.

The data used in this study was collected in May 2023 for grassland and August 2023 for wheatland using the UAV DJI Mavic 3 Multispectral (DJI, n.d.). These dates were based on the harvest months of the two crops. By capturing data from the fields just before they are harvested, a better visible contrast is present between damaged and non-damaged areas. The camera of UAV DJI Mavic 3 Multispectral (DJI, n.d.) consists of a 20 Megapixels RGB camera and single band cameras: green (G) 550nm, red (R) 650nm, and Red Edge (RE) 730nm with all a bandwidth of +- 16nm and a Near-infrared (NIR) camera of 860nm with a bandwidth of +- 26nm. The raw data collected by the UAV was in order of several Gigabytes per agricultural field. Each field was flown over five times, once in nadir position (e.g., the center of the UAV is pointed directly towards the surface) and the other four times, to capture the data needed for 3D generation, in the four different camera angle orientations with an oblique position of -45 degrees. The decision to use the oblique flight data for the 3D generation is due to the increased probability of capturing data that can be translated to ground points. The UAV flew at a height of 40m with a speed of 15m/s. These camera settings produced a ground sampling distance (GSD) (e.g., the distance between two consecutive pixel centers measured on the surface) of 2.3 cm.

The data from eight agricultural fields for both grassland and wheatland were randomly selected to be used for this study. These fields were split into two groups of four fields, representing the train (Tr#) and test data (Te#) used in this study (Figure 1). The train data was used for the hyperparameter tuning (i.e. a method to find optimal model parameters, refer to subsection 4.2 for an elaborate explanation) and training (e.g., learning a classifier the damage mapping, more information in subsection 4.2) of the object, pixel, and point cloud-based classifications. The test fields were used as independent data for the evaluation of the classifications. These fields in total cover an area of 19.5 hectares for grassland and 20.5 hectares for wheatland. Table S1 in Appendix 9.1 contains the total areas for the training and test fields separately.

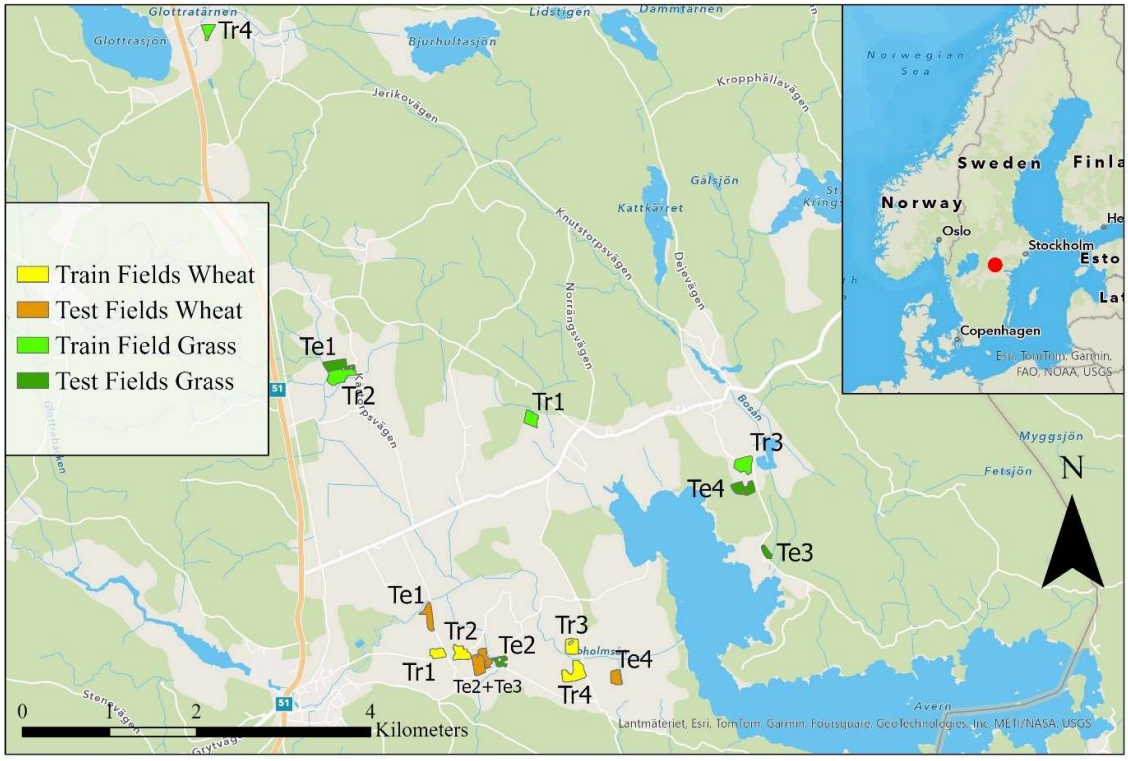

Figure 1: The study area Boo, Hjortkvarn Municipality, south-east of Örebro County, Sweden represented in the coordinate reference system WGS 84 / UTM zone 33N. In total UAV data from 8 grasslands and 8 wheatlands were collected. Along with the fields, it is indicated whether the field is used as train (Tr..) or test (Te..) data for grassland and wheatland separately. The data was collected in May 2023 for the fields belonging to Grass and in August 2023 for the fields belonging to Wheat using a UAV DJI Mavic 3 Multispectral (DJI, n.d.) that had a flight height of 40 meters and a speed of 15m/s.

PIX4Dmapper version 4.8.4. (Pix4D S.A., Prilly, Switzerland) was used to process the raw UAV data for the creation of the ortho-mosaics (i.e., UAV images of one field stitched together into one multilayer raster) containing the R, G, B, NIR, and, RE multispectral values, these reflect the bands of the ortho-mosaic. The raw images had an initial spatial resolution of 2.3 centimeters (GSD), using a bilinear interpolation the resolution of the ortho-mosaics was resampled to 5cm spatial resolution. The ortho-mosaics were in the coordinate

reference system WGS 84 / UTM zone 33N (i.e. EPSG: 32633). There was no radiometric calibration performed, as the fields were processed separately and were captured during similar weather conditions (i.e. sunny). Next to this, there was no need to calculate precise plant phenology metrics, therefore accurate reflectance values were of less importance to this study. However, each field's bands were normalized to a range between zero and one (i.e. based on each field's maximum and minimum values) to mitigate the potential effect of radiometric variations between the fields. See Tables S2-5 in Appendix 9.1 for the created ortho-mosaics of each field separately, only the RGB bands are shown for visualization purposes.

The creation of the UAV point clouds was also performed using the PIX4Dmapper version 4.8.4. LIDAR data (i.e. 3D data created from a sensor that actively fires pulses to the surface and recollects them to retrieve detailed height information) can provide high-quality data to create high-density point clouds that contain return information (i.e. information indicating whether points are first returns, meaning that they are ground points, or later returns, reflecting the other layered matter on the surface). As LIDAR data was not available for all of the eight wheat fields, it was chosen to create point clouds from the images collected in oblique positions (i.e. photogrammetry-derived point clouds). These point clouds are of a lower point density compared to LIDAR data-derived point clouds and do not contain return information. The average density of the created point clouds was 1350 points per cubic meter (m^3). Information from the quality reports (i.e. total area, average GSD, keypoint match density, average point cloud density, mean reprojection error) retrieved for each field from the PIX4Dmapper processing is shown in Tables S6 and S7 in Appendix 9.1).

Next to the UAV data, validation field data for each grass and wheatland field was manually collected using a Global Navigation Satellite System (GNSS) receiver achieving centimeter accuracy. These consisted of coordinates of damage locations. The data consisted of damages originating from wild boar but also from machinery, deer, badgers, drought, floods, insects, rocks, and plant disease. This data was used as independent validation data for the evaluation of damage locations as mapped by the classifications.

4.2 General Methodology

Three methods for mapping damages on a cm-level spatial scale in agricultural wheat and grasslands were created and compared. The three methods were chosen to reflect a wide spectrum of ways to utilize UAV data to map damages in agricultural grass and wheatlands. Two methods focused on analyzing 2D multispectral UAV data (i.e. the object-based and pixel-based classifications), and the third focused on 3D UAV data (i.e. the point cloud height threshold classification). Figure 2 shows an overview of the methodology used in this study which was entirely created in R-Studio version 4.4.0 (R core Team (2024)). Refer to Table S16 in Appendix 9.4 for an overview of all the packages utilized in the creation of the R-pipeline.

The 2D methods differed in the way the 2D UAV data was processed before it was used as input to the classifiers (more information in subsection 4.3.2.2) of the classifications. The pixel-based classification provided its classifiers with single-pixel information to classify each pixel in the ortho-mosaic. For the object-based classification, the ortho-mosaic was segmented into groups of pixels (subsection 4.3.2.1). The segments could provide additional information, such as contextual, morphological (e.g. related to the shape of the segment) and spatial data for the classifier to use (Blaschke, 2010; Hay & Castilla, 2008; van der Werff & van der Meer, 2008). In this way, the classifier did not just classify every pixel of the ortho-

mosaic (i.e. pixel-based classification) but whole objects (i.e. object-based classification). Via this, the 2D classifications were either done on pixels or objects, thereby an exploration of the influence of input data on the damage mapping in agricultural grass and wheatlands using 2D UAV data could be achieved.

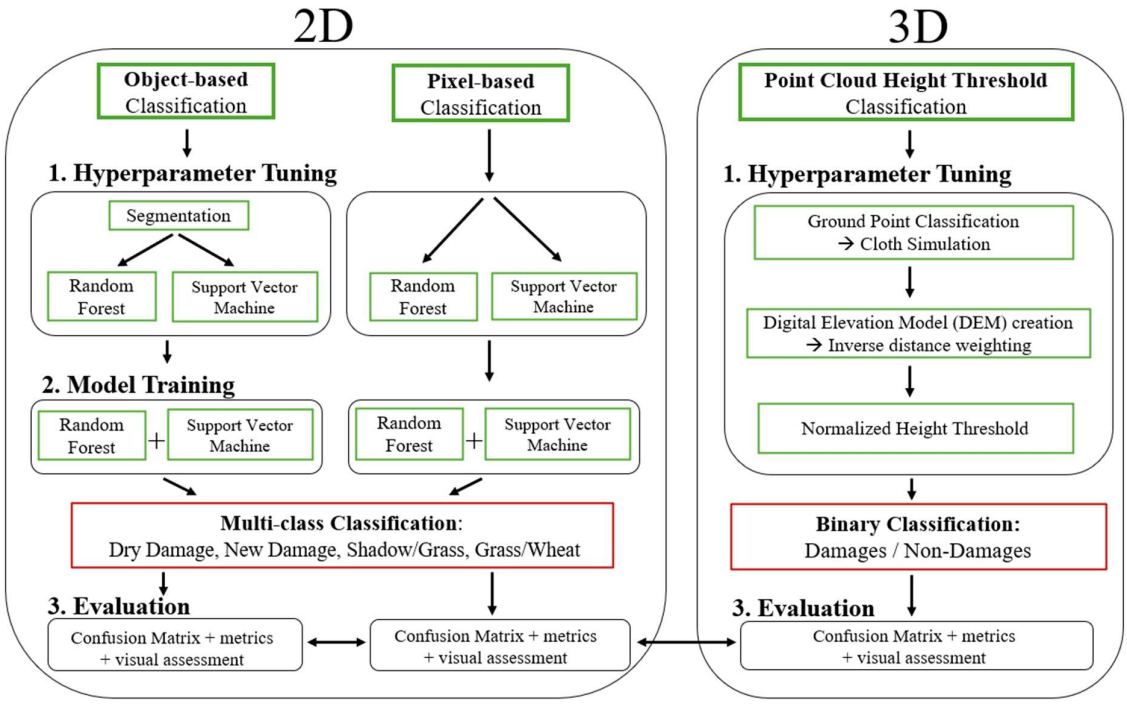

Figure 2. The workflow used for the mapping of damages in agricultural wheat and grasslands in this study. The workflow consists of three classifications divided into two 2-dimensional (2D) UAV-based methods and one 3-dimensional (3D) UAV-based method. The 2D-based method consisted of the components hyperparameter tuning, model training, and evaluation, and the 3D-based of the hyperparameter tuning and evaluation. The last step of the workflow consisted of the comparison of the retrieved evaluations of the classification methods. The data from this study was collected using a UAV DJI Mavic 3 Multispectral (DJI, n.d.) that had a flight height of 40 meters and a speed of 15m/s in May 2023 for grassland and August 2023 for wheatlands in Boo, Hjortkvarn Municipality, Sweden.

The 3D point cloud height threshold classification differs from the 2D methods, as it did not use the multispectral UAV data but used the 3D UAV point cloud, where the height of points in space was used to classify an area as damage or not. This method was based on the idea that damaged areas have a lower height than non-damaged areas (i.e. crops). A medium-density point cloud was used for this method (detailed explanation in previous section 4.1), this resulted in the method being applied only on wheatlands, as the height difference between grass and damaged areas in these photogrammetry-derived point clouds was not significant for the method to work on grasslands.

Both the 2D and 3D methods to map damages in agricultural grass and wheatlands were based on the notion of image classifications (Perumal & Bhaskaran, 2010). This means that specific units of the UAV data, pixels for pixel-based, objects of pixels (i.e. segments) for object-based, and 3D points for the point cloud height threshold classification, were assigned (i.e. classified) to a specific target class. These target classes are informational categories that

are of interest to the user and depend on the scale (i.e. cm-level) of the classification (Palaniswami et al., 2006). For the 2D classifications, it was chosen to have four target classes for both grass and wheatlands (i.e. multi-class classification, see Figure 2). The target classes were chosen with the interest of being able to distinguish damaged areas from other phenomena in the agricultural fields. When choosing target classes it was important that the pixel/segment values of the target classes were distinct from each other and that all instances (i.e. pixels/segments) of the same target class had similar pixel/segment values. This was necessary as the classifiers learn what specific pixel/segment values corresponded to what target class. By having the classes as homogenous as possible the risk of confusion between target classes gets lowered. Considering the high spatial resolution (i.e. cm-level) of this study, it was chosen to have two target classes representing damage. This was done to lower the heterogeneity within the damage target classes. Considering the other phenomena present in the agricultural fields, the target classes for grassland were set to dry damage (e.g., older damage therefore at data capturing bare land/dry soil), new damage (e.g. recent damages therefore wetter soil), shadow (e.g., areas blocked from sunlight), and grass. For wheatlands, the target classes were set to dry damage, grass, dark damage (e.g. recent damages containing destroyed wheat or wet soil), and wheat. As Figure 2 shows, the 3D point cloud classification was a binary classification, meaning that only two target classes, damage and non-damage, were considered during the classification. As the method was based on height differences between the wheat plants and the damaged area, target classes such as dark and dry damage (used in the 2D wheatland classifications) were non-distinguishable.

Both 2D methods were based on supervised classifications to map damages in agricultural grass and wheatlands. Supervised classification is a form of image classification that consists of training a classifier to predict the target classes, where the training data contains both the pixel/segment values and the associated target class. By the availability of the target class information, the classifier can learn what data corresponds to what target class and therefore learn how to predict new situations (e.g. where the target class is unknown). The 3D method used an unsupervised classification, which is another form of image classification. This form has the advantage that it does not require training data that contains the target classes to create the classification model. The point cloud height threshold method classifies the points in the point cloud into two unspecified classes, after which the correct target class has to be assigned to it (e.g., via a height threshold, refer to subsection 4.4 for a detailed explanation of the 3D point cloud height threshold classification method). Despite the advantage of the unsupervised classifications for not needing target class data, the 2D classifications made use of supervised classifications as the assignment of the target classes can become very difficult to automate when working with high-dimensional data (i.e. the bands of the ortho-mosaic, more information in subsection 4.3.1) and multi-class classifications.

Figure 2 shows that the two 2D methods consisted of three main components, hyperparameter tuning (1), model training (2), and evaluation (3). The 3D method consisted of 2 main components, the hyperparameter tuning (1) and the evaluation (3), as the model training (2) was not needed due to the method being an unsupervised classification. The first component, the hyperparameter tuning (1), was used to find optimal settings (i.e. hyperparameters) for each method's classification model. A hyperparameter tuning is a technique that automatically and systematically searches for optimal hyperparameters to optimize (i.e. tune) the model performance. Whether hyperparameters are optimal is checked upon a performance measure that is to be set by the user. Several algorithms exist to tune the hyperparameters of a machine learning model, such as Random Search (Bergstra & Bengio, 2012), Bayesian optimization (Snoek et. al., 2012), and gradient optimization (Chapelle et. al., 2002). These algorithms, among others, are good for situations where a lot of hyperparameters are present,

as they systematically search (or randomly in case of random search) in the hyperparameter value space for more optimal solutions. For this study, a full grid search was chosen, hereby all combinations of the provided hyperparameter values were assessed in performance. This was possible as only a few hyperparameters were present to tune in each of the methods (i.e. the pixel, object-based, and point cloud classifications). Refer to subsections 4.3.2.3, 4.3.3.1, and 4.4.3 for object-based, pixel-based, and point cloud classification methods respectively, for detailed information on the hyperparameters tuned and the performance measure used within the method. Once the optimal settings were found for each model, the 2D models could be trained (component 2), or in the case of the 3D method, were ready to perform the damage mapping after which the damage mappings were evaluated (component 3). Lastly, the classifications and the evaluations of the different methods were compared. The upcoming subsections will explain each of the methods in more detail, starting with the two 2D methods.

4.3. Object and Pixel-based Classification (2D)

4.3.1 Pre-processing

Pre-processing steps had to be performed on the 2D multispectral UAV ortho-mosaics (explanation in subsection 4.1) before it was ready to be used by the object and pixel-based classifications (refer to Table S16 in Appendix 9.4 for the used R packages). First, any data outside of the agricultural field itself (e.g., farmer's house or patches of forest) present in the ortho-mosaics were removed. It was chosen to add, next to the original five bands in the ortho-mosaics (i.e. R, G, B, NIR, and RE), a band containing the vegetation index Normalized Difference Vegetation Index (NDVI) and bands containing the Texture indices RGB/NIR mean and RGB/NIR variance. The decision to add a vegetation index was based on the expectation that it could provide valuable information for a classifier to use when discriminating between damaged areas, containing relatively more soil (lower NDVI), and crops (higher NDVI). Furthermore, texture was calculated as it was observed in the UAV images that texture differences existed between the target classes. It was therefore expected that the texture indices could increase the performance of the classifications.

The NDVI was calculated based on the NIR and R values and derived using the original formula (Formula 1) proposed by Rouse et. al. (1974).

$$NDVI = \frac{\text{NIR} - \text{R}}{\text{NIR} + \text{R}} \quad (1)$$

Where NIR is the Near-infrared and R the red wavelength bands.

Utilizing the R-package GLCMTextures (Ilich, 2020), the RGB (i.e. based on the data from the 20 Megapixels RGB camera) and NIR mean and variance texture values were calculated based on their Gray-Level Co-Occurrence Matrix (GLCM) (Haralick et al., 1973). This GLCM consisted of the quantized RGB and NIR values in the ortho-mosaic divided into 16 equal probability quantile bins (i.e. each bin contains an equal number of pixels). Using the GLCM (i.e., containing 16 gray levels), a moving window of five by three pixels was used to calculate the texture mean and variance values for the pixels in the ortho-mosaic. All the bands together (R, G, B, NIR, RE, NDVI, RGB mean, NIR mean, RGB variance, NIR variance) made a total of ten bands available for the 2D classifications.

Second, as mentioned in subsection 4.2, the 2D image classifications were based on supervised learning. This means that it was needed to create data containing the locations of target classes in the ortho-mosaics. Separate target class data had to be created for the training

and testing within the hyperparameter tuning (component 1 in Figure 2), the model training (component 2 in Figure 2), and the model evaluation (component 3 in Figure 2). These data are all separate datasets to avoid creating a bias in the evaluation results of the trained classifications for mapping damages in agricultural grass and wheatlands. By visual assessment of the ortho-mosaics, this target class data was manually created in QGIS 3.28.10-Firenze. For the 4 fields assigned as train fields (subsection 4.1 and Figure 1), 40 polygons per target class (i.e. 4 target classes so 160 in total) were randomly selected for the training of the classifier during the hyperparameter tuning (component 1) and another 40 per target class for the training of the classifier for the model training (component 2). Next to this, test data from these train fields was created for the evaluation of the hyperparameter tuning (component 1) and the evaluation of the trained model (component 3), these consisted respectively of 20 and 40 randomly selected polygons per target class. In addition to the test data created from the four train fields, test data was created from the four selected test fields (subsection 4.1 and Figure 1), thereby serving as completely independent test data. For these fields, 40 random polygons were selected per target class as test data. The data from the independent test fields were only used for the evaluation of the trained classifiers (component 3), not for the evaluation within the hyperparameter tuning (component 1). Table 1 shows the created train and test data.

Table 1. The number of fields and polygons used as train and test data for the classification and hyperparameter tuning. The polygons represent samples of the target classes Dry Damage, New/Dark Damage, Shadow/Grass, and Grass/Wheat. The number of fields and polygons displayed is available for grassland and wheatland separately.

What	**Amount of Fields**	**Amount of polygons**	**Area (m^2) Grassland**	**Area (m^2) Wheatland**
Train data classification	4	640	585	349
Test data classification	8	1280	452	740
Train data hyperparameter tuning	4	640	584	487
Test data hyperparameter tuning	4	320	240	219

The upcoming subsubsections will contain a detailed explanation of the workings of the object-based and pixel-based classifications.

4.3.2 Object-based Classification

The main package used for the object-based classification was the R package SegOptim (Gonçalves et. al., 2019). This package allows the performance of both segmentation (i.e. element 1 of the object-based model, see Figure 2) and classifier training (i.e. element 2 of the object-based model, see Figure 2) within the R environment for single and multi-class problems. Refer to Table S16 in Appendix 9.4 to see all R-packages used in this method.

4.3.2.1 Object-based: Segmentation

As explained in subsection 4.2, the object-based classification used objects of pixels (i.e. segments) as input to the classifiers. Therefore, the first element of the object-based classification model was the segmentation. The segments were created by merging pixels in the ortho-mosaic according to homogeneity criteria (Blaschke, 2010). The criteria of the

segmentation depend on the objective of the task. For this method, the objective was that a segment solely consisted of pixels belonging to one of the four target classes. As segmentation is a computationally heavy task, one out of the ten available bands was selected to perform the segmentation with. The band that was selected, was the band that had spectrally the largest separability ability between the different target classes. To assess this separability, the Jeffries-Matusita (JM) distance was used. The JM is a statistical separability criterion widely used to assess the separability of classes by spectral bands (Dabboor et al., 2014). The version of the JM by (Dalponte et al., 2013; Bruzzone et al., 1995) available via the R-package VarSel (Dalponte & Ørka, 2021) was used to perform the calculations. The separability value is given per band between each combination of target classes (e.g. New Damage vs Dry Damage, New Damage vs Shadow, Shadow vs Dry Damage, etc., six combinations in total). The JM score has a range between zero and two, where zero means non-separability, one medium separability, and two full separability. The band giving the maximum sum of the JM scores of all the combinations of target classes was used to perform the segmentation. The hyperparameter training data created from the four training fields was used for the calculations of the JM. The RGB mean texture band had the highest separability for both grass and wheatland with an average value of 1.08 and 1.04 respectively. Therefore the RGB mean band was used to perform the segmentation for both grass and wheatlands. Figures 3 and 4 show the histograms of the RGB mean bands for grass and wheatland. Refer to Table S8 in Appendix 9.2 for the JM scores per combination of target classes for the RGB mean and Table S9 Appendix 9.2 for the top 5 bands with the highest summed JM scores.

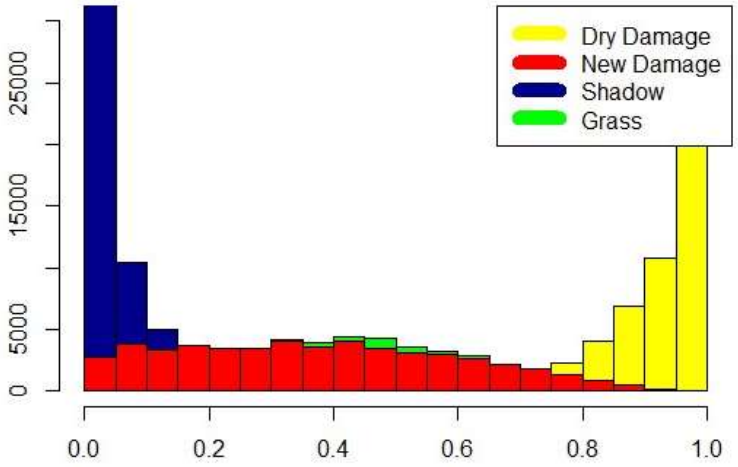

Figure 3. The histogram shows the separability ability of the RGB mean band between the different target classes for the grassland fields (i.e. Dry Damage, New Damage, shadow and Grass). The normalized spectral values are represented on the x-axis and the frequency is displayed on the y-axis.

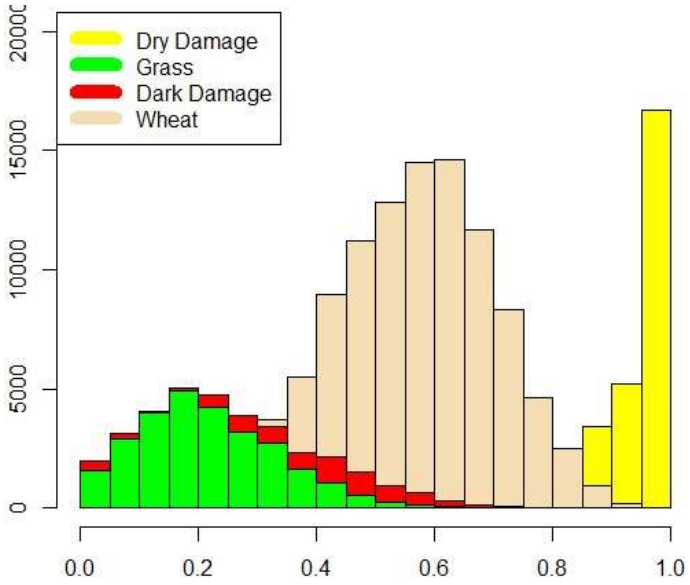

Figure 4. The histogram shows the separability ability of the RGB mean band between the different target classes (i.e. Dry Damage, Dark Damage, Grass and Wheat). The normalized spectral values are represented on the x-axis and the frequency is displayed on the y-axis.

For each of the created segments, the mean and standard deviation of the combined pixels from each of the ten input bands were calculated and the datasets containing the target classes (section 4.3.1, Table 1) were joined to their corresponding segment. Therefore, in total 20 bands (i.e. 10 mean and 10 standard deviation bands) per segment were available as input to the classifiers (second element). As the manually selected polygons only consisted of parts of target class areas (e.g. the polygon does not contain the full outline of one damage spot but a part of it), the attachment of the target classes was done by setting the inclusion threshold at 0.01% overlap with a segment. This was done with the assumption that each segment satisfies our objective of containing solely one target class. This means that when 0.01% or more of the segment was covered by a polygon, the segment was used as train/test sample and labeled to the corresponding target class of the polygon. Once this was done, the data was prepared to be used by the three components of object-based classification, namely the hyperparameter tuning (i.e. component 1, Figure 2), training of the classifiers (i.e. component 2, Figure 2), and the evaluation of the trained classifiers (i.e. component 3, Figure 2).

4.3.2.2 Object-based: Model Classifiers

The second element of the object-based classification model is the model classifiers. Random Forest (RF) and Support Vector Machine (SVM) were selected, among other reasons discussed in the paragraphs below, due to the promising results from previous studies that used the RF and SVM for related agricultural applications (Subsection 3.2). The following two subsections contain a detailed theoretical explanation of the RF and SVM classifiers.

4.3.2.2.1 Random Forest

The Random Forest algorithm used in this study was proposed by Breiman (2001) and consists of multiple decision trees that perform the prediction. Each decision tree has to learn how to predict the correct target class given the input data. To prevent the decision trees from learning similar information, randomness is incorporated in the training data used for the individual decision trees in both the features (e.g. the bands to use) and training samples

(what part of the created training objects). Given each tree's randomly selected features and training samples, the tree has to learn how to classify them into the target classes. This is done by splitting up the data into increasingly homogenous groups of target classes. The splits are based on conditional statements such as *'Data samples containing a spectral value higher than 200 for Red is split from the samples with a value for Red equal or lower than 200'*. The Gini Index calculation is used to find the optimal split. The Gini index represents the probability of misclassification after the split (Formula 2).

$$Gini\ Index = \sum_{i=1}^{c} P(\mathrm{i})^2 \qquad (2)$$

Where c is the number of classes (i.e. four) and P(i) is the probability of misclassification of class i (i.e. the amount of class i in the split divided by the total amount of instances in this split).

For four classes the maximum impurity split would have a Gini index of 0.75. A pure split, the preferred situation (e.g. where only samples of one target class are present in the group after the split), has a Gini index of 0. A split is made in such a way that the Gini index after splitting is as small as possible. Once every decision tree is trained, the trees together form the Random Forest. The classification by RF is made by selecting the target class that has the majority of the outcomes of all the decision trees.

The notion of Gini Index is also used for calculating each band's importance within the whole RF. To calculate each band's importance, the mean decrease of the Gini Index is calculated. This measurement indicates the influence of each band on the homogeneity of the nodes (groups of split samples) in the decision trees that make up the RF. The bands that eliminate the most impurity in nodes have the highest mean decrease values. The higher the mean decrease value is, the higher the importance of the band (Martinex-Taboada and Redondo, 2020). E.g. the Gini Index shows what bands contain most of the information needed to map damages in agricultural wheat and grasslands.

The RF classifier was chosen for the 2D classifications because it can classify categorical target classes and can handle datasets with high dimensionality (e.g., 10 bands in the ortho-mosaics). Next to that, because of the multiple decision trees, a RF can learn complex, non-linear relationships between the input bands and the target classes. Another advantage of the use of multiple decision trees is that a RF is less prone to overfitting. Lastly, a RF can provide information about the importance of specific bands for doing the classification. This information can give valuable insight and understanding into the application of damage mapping in agricultural grass and wheatlands (AIML, 2023).

4.3.2.2.2 Support Vector Machine

The foundation of the SVM was introduced by Vapnik (1998). A SVM searches and defines a higher dimensional space where the samples belonging to different target classes are maximally separable (Scholkopf & Smola, 2018). The separation is defined by a hyperplane (i.e. decision boundary separating the samples of the target classes) in the higher dimensional space. The original SVM algorithm was created for linear class discrimination (i.e. two class/ binary classification) but later adaptations have transformed the algorithm into one that has a non-linear working. SVM uses a concept named Structural Risk Minimization (Vapnik, 2013), where the hyperplane is created in such a way that the distances (i.e. margins) between the closest sample of each target class to the hyperplane are maximized. Next to this, is the complexity of the model taken into account, thereby preferring simplistic models over complex models. In these ways, the chance of overfitting gets reduced (e.g. generalization

improves). The coefficients used for the vectors that make up the SVM hyperplane can be used as a measure of importance. By squaring up all the coefficients per band, an estimation can be retrieved of how much influence a band has on the hyperplane used for classification.

Next to the generalization characteristics of a SVM, another reason to investigate the performance of the SVM classifier in this study is that it works effectively with high-dimensional data. Lastly, did, among others, Najah et al. (2011) conclude that the SVM is computationally more efficient than classifiers like ANN and that a higher precision can be retrieved with less computational complexity.

4.3.2.3 Object-based: Hyperparameter Tuning

To optimize the performance of the object-based classification models (i.e. segmentation + RF classifier and segmentation + SVM classifier, Figure 2), a hyperparameter tuning (component 1, Figure 2) was performed on the hyperparameters for the segmentation (refer to subsection 4.2 for the explanation of hyperparameter tuning). The segmentation function provided by the R-package SegOptim required three user-defined hyperparameters, which were the spectral range, spatial range and the minimum segment size (Michel et. al., 2015). The spectral range indicated the maximum difference of the spectral values in the feature space that was allowed to be within one segment, the spatial range was the maximum distance in the neighborhood and the minimum segment size was the minimum amount of pixels a segment should contain to be considered.

The provided value ranges used in the hyperparameter tuning were based on theory, trial-and-error explorations on the study's data, and default values provided by the SegOptim R-package (Gonçalves et. al, 2019). Refer to Table 2 for the chosen value ranges per hyperparameter. As all the data was normalized to a range between zero and one, the spectral range values indicated that variations of 1%, 2%, 3% or 4% in spectral value within one segment were permissible. When the value for the spectral range is too high, the segments become too heterogeneous, thereby increasing the chance that segments did not satisfy the objective (i.e. one segment only contains pixels belonging to one target class). The spectral range was set to a maximum of 30 pixels, mainly due to the time limitations present in this study, as computation time increased drastically when the search space was set to even larger areas. The minimum size of the segments was set to a maximum of 25 pixels, to prevent small target areas (e.g. small damage spots) from being removed from the analysis. A minimum was set to 15 pixels to prevent situations where segments were created that consisted of just one to very few pixels. This situation had to be prevented as the performance could become artificially high (or low). This happens because one damage spot gets represented by a lot of different segments. This then results in test data that refers to one damage spot, is represented by multiple one to few pixel(s) objects (see Figure S1 Appendix 9.2). When this damage spot is predicted correctly by all the segments representing this object, this one damage spot counts as many correct predictions, making the classification look like it performs better than it actually does. This is a situation that has to be avoided.

For each combination of hyperparameter values, the segmentation (i.e. element 1) and the training of the machine learning classifier (i.e. RF and SVM, element 2) were performed and evaluated in optimality using the Kappa coefficient (refer to subsubsection 4.3.2.5 for explanation of this performance metric). The hyperparameter tuning was done for both the RF and SVM classifiers separately using the same created hyperparameter training and test datasets (subsection 4.3.1, Table 1).

Table 2. Hyperparameter tuning - full grid search for the segmentation function that is part of the object-based classification. These values were used for the hyperparameter tuning of both the models containing Random Forest and Support Vector Machine. The spectral range, spatial range, and minimum size are hyperparameters that need to be set in the segmentation method provided by R-package SegOptim (Gonçalves et. al., 2019).

Hyperparameter	Values
Spectral range	0.01, 0,02, 0.03, 0.04
Spatial Range	10, 20, 30
Minimum size	15, 20, 25

4.3.2.4 Object-based: Model Training

After the optimal hyperparameters were retrieved, the actual model could be trained (component 2, Figure 2). The RF and SVM classifiers were trained using a 10-fold cross-validation. This means that the created training data for the classification model (information in subsection 4.3.1 and Table 1) was split up into 10 equal randomly sampled parts (i.e. folds). The model was trained 10 times using each time 9 different folds for training and the leftover fold for calculating the expected performance (i.e. Kappa and Overall Accuracy, explanation in subsection 4.3.2.5) of the trained classifier. The expected performance gives an indication of how well the trained classification model will perform on the test data (subsection 4.3.1 and Table 1). By training the model 10 times using different parts of the train data, a more robust estimation of the expected performance could be retrieved.

SegOptim used the RF algorithm provided by the R-package RandomForest (Liaw & Wiener, 2002) and the SVM algorithm (Boser et al., 1992; Vapnik, 1998) provided by the R-package e1071 (Meyer et al., 2017). The RF consisted of 250 decision trees that were trained using 4 randomly selected bands as candidates for the splits (refer to subsection 4.3.3.1 for an elaborate explanation). Refer to Table S16 Appendix 9.4 to see all R-packages used in this object-based classification method.

4.3.2.5 Object-based: Model Evaluation

Once the models of the object-based classification were trained, the models could be evaluated using the created classification test data (subsubsection 4.3.1 and Table 1). The following performance evaluation metrics were calculated for both the pixel and object-based multi-class classifications.

- ***(Overall) Accuracy of the whole classification.***
  The overall accuracy is the total number of correct classifications divided by the total amount of classifications. The range is between zero and one, where one means 'perfect' classifications and zero means all classifications are wrong.
- ***Kappa coefficient of the whole classification.***
  The kappa coefficient takes into account the correct classifications that you would have by chance (Ensrud, 2021). Hereby the expected agreement (the agreement at chance level) is subtracted from the observed agreement (overall accuracy) and divided by (1 – expected agreement). A kappa coefficient of zero reflects that the classification is not better than chance and a coefficient of one means 'perfect' classification (no classification mistakes made according to the test data).

- ***F1 Score of the whole classification***
  This score is based on a trade-off between how many of the present target class instances are detected and how many of the classifieds are correct. Given the fact that four target classes were present, the average value of each of the target class's F1 scores was calculated and used as the F1 score.
- ***User's accuracy per target class.***
  The user's accuracy of a target class reflects how many of the pixels that were classified as this class are actually this target class. This accuracy is 100% when all the classifications predicted as this target class belong to this target class.
- ***Producer's accuracy per target class.***
  The producer's accuracy per target class reflects how many of the pixels belonging to a target class are also classified as this target class. This accuracy is 100% when all the pixels belonging to this target class are also predicted as such.

4.3.3 Pixel-based Classification

The pixel-based classification classifies each pixel into one of the 4 target classes. The pixel-based classification is faster, as the data after the pre-processing steps (subsection 4.3.1) could directly be used to train and test the RF and SVM classifiers (subsection 4.2 and Figure 2). The main package used for the pixel-based classification was the R-package Caret (version 4.47) (Kuhn, 2008). This package enables hyperparameter tuning within the training of the classifiers, thereby speeding up the computational running time. Refer to Table S16 in Appendix 9.4 for an overview of all R-packages used in this method.

4.3.3.1 Pixel-based: Hyperparameter Tuning

For the training of the RF 1000 decision trees were trained. The number of bands that were randomly sampled as candidates to create the splits in the decision trees of the RF was optimized during the hyperparameter tuning (component 1, Figure 2). This hyperparameter decides from the bands that are available for the decision tree, how many of those are randomly selected to be considered to perform the split on. This is done to prevent the decision trees from becoming too dependent on one or a few specific bands to perform the classification, thereby increasing the robustness of the trained classifier. This hyperparameter is a number between two and the total number of bands used for the classification (i.e. 10), therefore the tuning grid was set to the values 2, 4, 6, 8, and 10.

For the SVM classifier (theoretical explanation in subsection 4.3.2.2.2), the SVMLinear3 (Helleputte et al., 2023) was used. This decision was made as a linear kernel, which requires less computational capacity than a non-linear kernel (e.g., Radial Basis Function), was expected to retrieve good classification performances. The hyperparameter tuning (component 1, Figure 2) was set to find the optimal cost parameter and loss function. The cost parameter reflects how much a sample is penalized for being within a close margin between the different target classes. If the cost is too low, there is a chance of overfitting (i.e. noise point have a large influence on the classification) and if the cost is too high the model can become too generalized, it was therefore of importance to tune this parameter. The tuning grid was set to 0.2, 0.5, 1 and 1.5. The loss function is used to measure the performance of the SVM hyperplane in classifying the target classes and is used to adapt the hyperplane to optimality (i.e. lowest total loss). The two loss functions that were used in the hyperparameter tuning were L1 and L2, where L1 is a linear sum of the misclassifications and L2 is the sum

of squares of the misclassifications.

4.3.3.2 Pixel-based: Model Training and Evaluation

The training of the models (component 2, Figure 2) for the pixel-based classifications was done using a 10-fold cross-validation (see subsection 4.3.2.4 for an explanation). The same model evaluation metrics were used as was used for the object-based classification (see subsection 4.3.2.5).

4.4 Point Cloud Height Threshold Classification (3D)

Next to the two 2D methods, a point cloud height threshold classification was studied, a method based on the 3D UAV data. The main R-package used for this method was lidR (Roussel et al., 2020 and Roussel and Auty, 2024). Refer to Table S16 Appendix 9.4 for an overview of all R-packages used in this method.

4.4.1 Pre-processing Point Cloud

Before the photogrammetry-derived point clouds (subsection 4.1) could be used for the classification, the following preprocessing steps were performed.

First, any data outside of the agricultural field itself present in the point cloud were removed.

The second preprocessing step was related to noise removal. The photogrammetry-derived point clouds available for the wheat fields contained outliers originating from various sources (e.g. birds, wind, processing errors, sensor errors) that had to be removed as they could influence the analysis. The Noise Segmentation Algorithm was utilized from the R-package lidR (Roussel et al., 2020; Roussel & Auty, 2024) to perform this noise removal. This algorithm is based on the Statistical Outliers Removal method, first described in the R-package Point Cloud Library (Rusu & Cousins, 2011). The algorithm uses the mean distance and standard deviation of points to their *k* closest neighbors to define a maximum distance threshold. Visual performance assessment of the noise removal resulted in the decision to use the 15 closest neighbors of each point to calculate the mean distance and standard deviation for. The maximum distance threshold is based on Statement (1), hereby it was chosen that three times the standard deviation is allowed on top of the mean distance for being considered a non-noise point. All points with a distance larger than this threshold are classified as noise and removed from the point cloud.

IF *Point distance > Mean distance + 3 * standard deviation* ***THEN*** *noise* statement (1)

The Point Cloud Height Threshold method is based on unsupervised learning and does not require training data. However, to be able to test the performance of the classification during the hyperparameter tuning (component 1, Figure 2) and the model evaluation (component 3, Figure 2), test data was created from the manually collected validation field data (subsection 4.1). This validation field data originally contained only GNSS coordinate points of the locations of the damages. Therefore, these coordinates had to be transformed into polygons containing the outline of the damage spots. Using QGIS 3.28.10-Firenze, the ortho-mosaics were visually assessed to manually digitize polygons at the indicated damage locations. 40% of the created validation test data from the train fields (subsection 4.1 and Figure 1) was used for the hyperparameter tuning, and the other 60% of the train fields and 100% of the created validation test data from the test fields (subsection 4.1 and Figure 1) were used for the model evaluation (component 3, Figure 2).

4.4.2 Point Cloud Classification - Model

Figure 2 shows that the 3D-based method consisted of a model with 3 elements, the Ground Point Classification, the Digital Elevation Model (DEM) creation, and the Normalized Height Threshold. The main idea of the method was to retrieve the height above the surface (i.e. normalized height) for all the locations in the wheatfields, whereafter a height threshold could be applied to classify the locations in damage and non-damage. To be able to retrieve the normalized heights, the elevation of the surface in the form of a digital elevation model (DEM, i.e., representation of the topographic bare-earth surface) had to be calculated. First, the ground points within the photogrammetry-derived point clouds (element 1) had to be identified, after which a DEM could be created via interpolation of the ground points (element 2). After the DEM is subtracted from the heights of the points in the point cloud, the normalized heights could be retrieved and the height threshold for classification (element 3) could be applied. The upcoming subsections will contain an elaborate explanation of each of the three elements of the model used in the 3D-based method.

4.4.2.1 Ground Point Classification

The first element of the model consisted of the classification of the ground points within the denoised point cloud to be able to calculate the DEM. With this, the decision was made to consider damage (e.g. tracks, wild-boar rooting places) as matter that has a negative height (i.e. the height of damage in the form of pits in the ground is not considered as ground elevation height). This means that it was aimed to not obtain classified ground points in the damage locations. The topography of the fields was mostly flat with no steep slopes.

The algorithm used for ground point classification is based on the method developed by Zhang et al. (2016). It consists of a cloth simulation where the point cloud is inverted and a cloth is fitted using the notion of gravity to cover the surface (see Figure 6). A main advantage of this method is that there are only a few, easy-to-understand, hyperparameters to be determined. As this research aimed to create a method for the mapping of damages in agricultural lands that is as automated as possible, this robust with few parameters method was preferred over the ground classification methods that use morphological filters. These methods obtain good accuracies, but the parameters have to be adjusted for each local scene, which made them less useful for this study (Mongus & Žalik, 2012).

The first hyperparameter needed for the cloth simulation is the resolution of the cloth. This value is often set to be around the average distance of the points in the considered point cloud which was for this study 5 centimeters. The cloth resolution together with the area of your point cloud defines the number of grid particles the cloth consists of. These grid particles are movable in height during the simulation. At each iteration, gravity simulation changes the height of the movable cloth particles. How the gravity simulation moves the particles, is dependent on a second hyperparameter which is the rigidness of the cloth. The rigidness value is dependent on the type of terrain. As this study worked with agricultural lands, the value was set to 3, this specific value reflects terrains that are more or less flat with no steep slopes. At each iteration, after the gravity simulation is performed on the cloth particles, the height of each particle is checked for being lower than its nearest point in the point cloud. If this is the case, the particle's position is set to the height of that point and is unmovable by the gravity simulation from that moment on. After a maximum number of iterations are performed (i.e. 500) the gravity simulation is stopped. At this moment, the points in the point cloud that have a distance to their closest cloth particle that is smaller than the classification threshold are classified as ground points. This classification threshold is the third

hyperparameter and was tuned in the hyperparameter tuning (for details on the hyperparameter tuning refer to subsection 4.4.3).

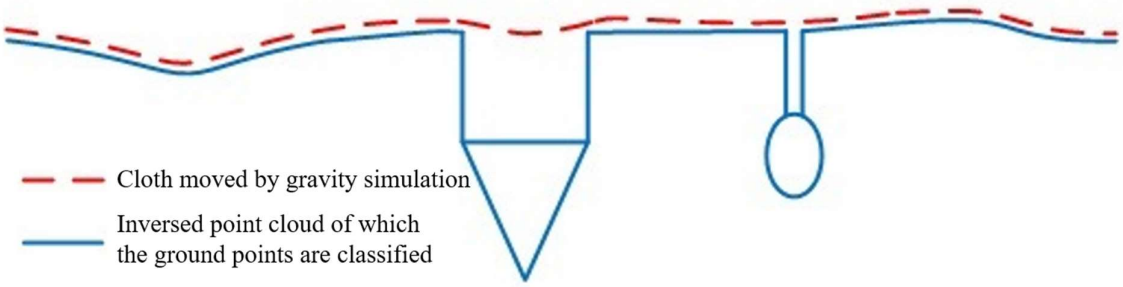

Figure 6. Simplified illustration (modified from Zhang et al., 2016, Figure 1) of the workings of gravity simulation on the cloth within the cloth simulation that was used to classify ground points in the photogrammetry-derived point cloud. The measurement represents the 180-degree turned point cloud on which a cloth simulation is performed. Points in the point cloud are classified as ground points when their distance to the cloth is smaller than a user-set threshold.

4.4.2.2 Digital Elevation Model Creation

The second element of the model consisted of the creation of the DEM. An interpolation had to be performed between the classified ground points to obtain the DEM of each of the wheat fields. For this, the Inverse Distance Weighting (IDW) method (Schut, 1976) was used which was available via the R-package Lidr (Roussel et al., 2020; Roussel & Auty, 2024). This method is a simplistic method that creates realistic DEMs, also on the edges of the study area. As the point clouds contained areas with missing points and the DEM had to be created including all the edges of the wheatfields, other methods like Kriging (Journel & Huijbregts, 1978) or Thiessen polygons (Longly et al., 2011) were less suitable.

The main idea of the IDW method is that all elevation points are related, whereas near points are more related to each other and should therefore have more influence on the estimated interpolated elevation value. This relatedness is inversely proportional to the distance from a specific point to each other point considered for interpolation. The power of the proportional is a hyperparameter and was tuned in the hyperparameter tuning. As well as the hyperparameter reflecting the amount of neighboring known elevation points that will be taken into account to perform the interpolation (for details on the hyperparameter tuning decisions and search grid refer to subsection 4.4.3). The resolution of the interpolated DEM raster was set to 10cm.

4.4.2.3 Normalized Height Threshold

The third element of the model consisted of applying a height threshold on the normalized height values. To retrieve the normalized height values at each location in the wheatfields, the created DEM was subtracted from the original point cloud. The normalized point cloud was then rasterized by taking the minimum height values and a resolution of 10cm to avoid retrieving missing data values within the created raster. The minimum height was selected, as it prioritized low height values that could potentially represent damage pits (e.g., areas with minus height values). Once the raster with the normalized height values was retrieved, the raster could be classified into the binary classification of Damage and Non-Damage. The threshold was set to have cells classified as Damage when they were containing a height

equal to or below zero. The cells with a normalized height value of above zero were classified as Non-Damage. This threshold was chosen as it was assumed that non-damaged wheat plants have a positive height and damages have negative heights as they are pits in the ground.

4.4.3 Point Cloud: Hyperparameter Tuning

A hyperparameter tuning (component 2, Figure 2) using full grid search was performed to retrieve the optimal hyperparameter value for the classification threshold for the ground point classification algorithm (subsection 4.4.2.1) and for the hyperparameters, power of the proportional distance and the number of neighbors, used for the Inverse Distance Weighting Interpolation of the classified ground points (subsection 4.4.2.2). See Table 3 for the hyperparameter values that were evaluated for optimality. The aim to have points in damage pits not classified as ground points was taken into account by having higher values for the number of neighbors used in the interpolation. Using a high number of neighbors ensures that the heights of points in damage pits that got classified as ground points are smoothed out during the interpolation. The test data, created from the validation field data of the train fields (subsection 4.4.1) was used to measure the performance of each of the hyperparameter combinations. The dice coefficient was used as the performance measure and indicated the pixel-wise agreement of the test data with the classified data (see formula 3).

$$Dice\ Coefficient = \frac{|X \cap Y|}{|X| + |Y|} \quad (3)$$

Where $X \cap Y$ is the amount of pixels that both the test data and the classification have as damage. $|X|$ is the amount of pixels containing damage in the test data and $|Y|$ is the amount of pixels within the damage areas of the test data that are classified as damage by the classification.

Table 3. The hyperparameter values used for the grid search for the hyperparameter tuning of the 3-dimensional UAV-based method which performs a point cloud height threshold classification. The classification threshold is used in the cloth simulation and represents the distance that points in the point cloud can have to the cloth particles to be classified as ground points. The power of proportional distance and the number of neighbors are hyperparameters used for the interpolation method called Inverse Distance Weighting (IDW), which is performed on the identified ground points to create a digital elevation model. The IDW method and the cloth simulation method provided by the R-package Lidr (Roussel et al., 2020; Roussel & Auty, 2024) were used.

Hyperparameter	**Values**
Classification threshold	0.0001, 0.000025, 0.00005, 0.000075, 0.00001
Power of proportional distance	1, 2
Number of neighbors	20, 50, 80

4.4.4 Point Cloud: Model Evaluation

After the optimal hyperparameters were retrieved, the classifications were performed for all fields and evaluated with the test data created from the validation field data for the classification evaluation (subsection 4.4.1).

4.5 Validation Field Data Evaluation

As shown in component 3 in Figure 2, the performance of the three classification methods (i.e. object-based, pixel-based, and Point Cloud Normalized Height classification) were compared. To be able to compare the three classifications, equal test data was required. It was therefore decided to use all the validation field data, preprocessed similarly as described in subsection 4.4.1, as test data for the grasslands and to use the test data created from the validation field data for the testing of the classification in the 3D method (subsection 4.4.1) as the test data for the wheatlands. This test data contains only information on damage areas, therefore the 2D methods, that classified into 4 target classes, had to be reclassified to a binary classification of Damage and Non-Damage. For the grasslands it was decided to merge the Dry Damage and New Damage into one Damage target class, to remove the Shadow target class from the analysis (i.e. set to NoData), and to reclassify the Grass target class as Non-Damage. For the wheatlands, the Dry Damage and Dark Damage were merged into one Damage target class, the grass was removed from the analysis and the Wheat target class was reclassified as Non-Damage. Next to using this test data to compare the three methods, difference maps between the different classification methods were created.

5. Results

5.1 Object and Pixel-based Classification Results (2D)

5.1.1 Object-based Classification Results

5.1.1.1 Object-based: Hyperparameter Tuning Results

Refer to Table 4 for the hyperparameter tuning results of the object-based method. Table S10 in Appendix 9.2 shows the top 5 hyperparameter combinations for the object-based classifications. For grass and wheatlands, it shows the higher the spatial range the lower the performance. The same pattern but weaker could be observed for the spectral range. Lastly, there was a preference for medium to high minimum segment sizes.

Table 4: Results of the hyperparameter tuning for both the Random Forest and Support Vector Machine classifier using Object-Based classification. The spectral range, spatial range, and minimum size are hyperparameters that had to be set in the segmentation method provided by R-package SegOptim (Gonçalves et. al., 2019).

Hyperparameter	**Grassland RF**	**Grassland SVM**	**Wheatland RF**	**Wheatland SVM**
Spectral Range	0.01	0.04	0.01	0.02
Spatial Range	10	10	10	10
Minimum Size	20	25	20	20
Kappa	0.97	0.96	0.97	0.97

5.1.1.2 Object-based: Band Importance

Refer to Table 5 for the band importance values obtained by the object-based classifications. It can be observed that the RF classifier had all the original sensory bands (R, G, B, NIR, RE) in the top 10, whereas this was not the case for the SVM as more derived texture values and standard deviations were of importance. A comparison of the importance values for the RF between grass and wheatland shows more derived texture values in the top 10 for wheatland. The importance values for the SVM between the grass and wheatland show little difference.

Table 5. Top 10 retrieved band importance values of the object-based trained RF (Mean decrease GINI) and SVM (squared coefficients) models with the optimized hyperparameters for both grass and wheatlands. The bands are either the mean values of the band in the segments, represented by the original band names, or the standard deviation (std) of the band values in the segments, represented by the band name + std.

Grassland Object-RF		**Grassland Object-SVM**		**Wheatland Object-RF**		**Wheatland Object-SVM**	
Blue	592.56	RGB mean std.	1057.70	Blue	557.28	RGB mean std.	2886.10
Green	294.52	RGB mean	1037.54	NDVI	517.92	RGB var. std.	2677.20
RGB mean	257.40	RE std.	1028.43	RGB var.	489.83	NIR mean std.	2540.76
NIR	227.83	NIR var. std.	1019.24	Red	414.85	Re std.	2494.02
Red	218.17	NIR mean std.	1011.40	RGB mean	412.45	NIR var. std.	2424.49

RE	212.31	NIR std.	1002.81	Green	395.19	Nir std	2283.70
NDVI	147.66	Green std.	989.96	Nir mean	392.81	Green	2240.92
NIR	135.83	NDVI	979.33	Nir	235.42	Red std	2107.77
RGB var.	97.76	NIR variance	978.34	RGB mean std.	220.51	RGB mean	2072.85
Blue std	89.36	Red std.	972.69	RE	207.53	Red	1966.26

5.1.2 Pixel-based Classification Results

5.1.2.1 Pixel-based: Hyperparameter Tuning

The pixel-based method results had, for both grass and wheatlands, the same optimal hyperparameters. These were 4 candidate bands used at the split by the RF and for the SVM the cost value 1.5 and the loss function L2. Figures S2:S5 in Appendix 9.2 show the behavior of the performance for the different pixel-based RF and SVM hyperparameter combinations. The results from the random forest classifier (Figures S2 & S4) show that a higher and lower number of predictors than 4 lowered the performance. The SVM results (Figures S3 & S5) show that the higher the cost, the higher the performance of the classification and that the L2 loss (i.e. squared losses) in general retrieved better performances.

5.1.2.2 Pixel-based: Band Importance

Refer to Table 6 for the band importances for the pixel-based classifications. The table shows that the derived texture RGB mean band was most important for each of the classifications. It can be observed that the SVM had similar bands as important for both grass and wheatland. These are after the texture band RGB mean, mainly the original UAV sensory bands (i.e. R, G, B, NIR, and RE). The RF had more texture-derived bands (e.g., NIR mean and RGB variance) in the top 5.

Table 6. Top 5 retrieved band importance values of the pixel-based trained RF (Mean decrease GINI) and SVM (squared coefficients) models with the optimized hyperparameters for both grass and wheatlands. The bands are either the mean values of the band in the segments, represented by the original band names, or the standard deviation (std) of the band values in the segments, represented by the band name + std.

Grassland Pixel-RF		**Grassland Pixel-SVM**		**Wheatland Pixel-RF**		**Wheatland Pixel-SVM**	
RGB mean	57074.83	RGB mean	397.12	RGB mean	43601.82	RGB mean	396.65
Blue	29386.21	Red	380.57	RGB var.	20699.88	Red	382.80
NIR mean	25410.60	Green	378.19	NIR mean	18291.48	Blue	373.26
Green	24627.49	Blue	373.21	Blue	13549.96	NDVI	357.67
Red	18757.84	RE	315.19	Red	11712.89	Green	344.60

5.1.3 Object & Pixel-based Classifier Train and Test Results

Tables S11 and S12 in Appendix 9.2 show the average scores for the kappa and overall accuracy from the 10-fold cross-validation training for the object and pixel-based RF and SVM classifier methods. All scores are above 84% for both grass and wheatland. The classifiers' test results are, for readability, split up into grassland and wheatland subsections.

5.1.3.1 Grassland: Object & Pixel-based Classifications Test Results

5.1.3.1.1 Grassland: Confusion Matrices and Test Evaluation Metrics

Tables 7 and 8 contain the evaluation results of the object and pixel-based RF and SVM classifications for the grasslands. The confusion matrices (Table 7) show that all classifications have instances where Dry Damage was confused with New Damage area, both ways. Next to this, they all have several instances where True Shadow was confused as New Damage, this is to a lesser extent the other way around. The evaluation metrics (Table 8) of the Object-based RF are lower than the other classifications, which all have similar values for the evaluation metrics. The highest accuracies are retrieved by the pixel-based RF classification.

Table 7. Confusion matrices containing the average values of the confusion matrices obtained from the test data origination from the separate fields used in the evaluation of the RF and SVM object-based and Pixel-based classifications for damage mapping in grassland. The numbers represent pixels.

Object-Based RF	**True Dry Damage**	**True New Damage**	**True Shadow**	**True Grass**
Pred Dry Damage	**3381**	328	123	24
Pred New Damage	232	**5837**	565	126
Pred Shadow	102	174	**4772**	165
Pred Grass	1	75	165	**6558**

Object-Based SVM	**True Dry Damage**	**True New Damage**	**True Shadow**	**True Grass**
Pred Dry Damage	**3544**	296	82	41
Pred New Damage	96	**5831**	305	124
Pred Shadow	72	177	**5053**	40
Pred Grass	2	110	184	**6667**

Pixel-Based RF	**True Dry Damage**	**True New Damage**	**True Shadow**	**True Grass**
Pred Dry Damage	**3605**	198	1	21
Pred New Damage	49	**5919**	364	11
Pred Shadow	59	272	**5187**	39
Pred Grass	3	26	72	**6802**

Pixel-Based SVM	**True Dry Damage**	**True New Damage**	**True Shadow**	**True Grass**
Pred Dry Damage	**3591**	235	3	17
Pred New Damage	60	**5607**	311	10
Pred Shadow	57	183	**5206**	54
Pred Grass	7	53	104	**6793**

Table 8. Evaluation metrics were obtained by the object and pixel-based RF and SVM classifications on grassland. Accuracy, Kappa and F1 are based on the four-class classification performances. The producer and user accuracies are based on the performance of the Dry and New Damage target classes taken together as one damage class. Std. shows the standard deviation of the evaluation metric.

Grassland Evaluation Metrics	**Object-based RF**	**Object-Based SVM**	**Pixel-Based RF**	**Pixel-Based SVM**
Accuracy	0.91	0.93	0.95	0.94
Accuracy Std.	0.02	0.02	0.04	0.03
Kappa	0.86	0.90	0.93	0.91
Kappa Std.	0.03	0.04	0.05	0.04
F1	0.88	0.91	0.94	0.92
F1 std.	0.08	0.06	0.03	0.04
Damage Producer's acc	0.87	0.93	0.94	0.94
Damage Prod. Acc. Std.	0.08	0.07	0.04	0.05
Damage User's acc	0.84	0.91	0.94	0.94
Damage User Acc. Std.	0.09	0.09	0.09	0.04

5.1.3.1.2 Grassland: Classification Output (2D-methods)

Figure 7 shows the damage mappings resulting from the object and pixel-based classifications for field *test4* (Figure 1) for the grassland. This field had the median amount of agreed classifications (see Table S13 Appendix 9.2) for both the object and pixel-based classifications. It can be observed that similar locations of damage spots were found by the object and pixel-based classifications but that the amount of classified damage pixels at a location was a lot higher for the object-based classification.

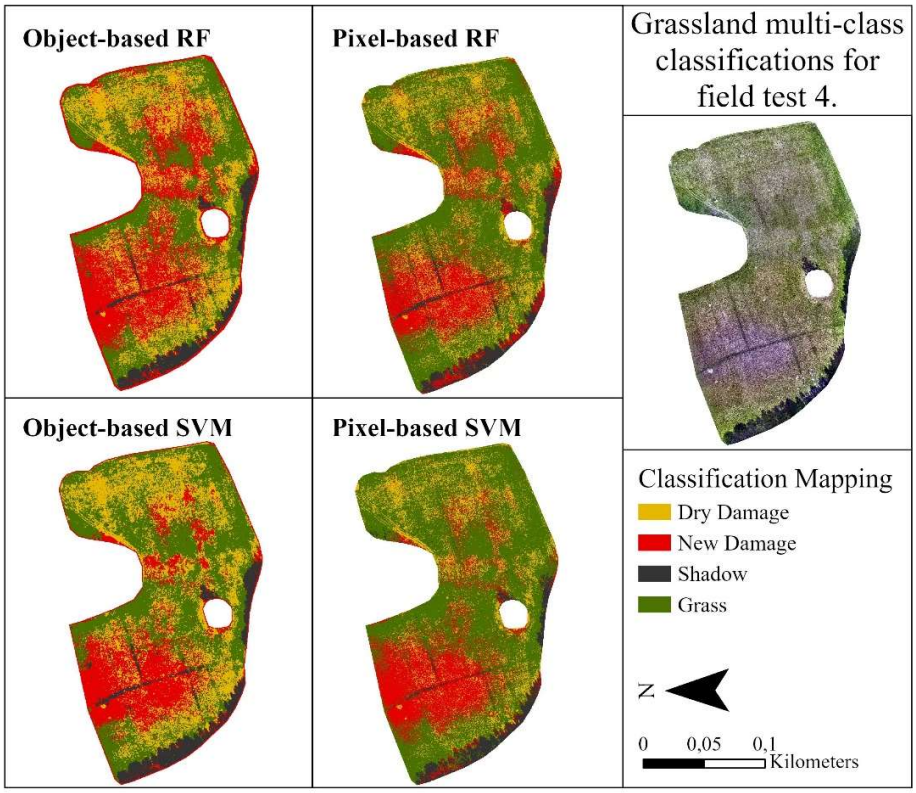

Figure 7. Visualization of the four-class classifications of the object and pixel-based RF and SVM classifications for grass field test 4 in coordinate reference system WGS 84 / UTM zone 33N. The data was collected in May 2023 in Boo, Hjortkvarn Municipality, Sweden using a UAV DJI Mavic 3 Multispectral (DJI, n.d.) that had a flight height of 40 meters and a speed of 15m/s.

5.1.3.1.3 Grassland: Damages Counts (2D-methods)

Table 9 shows for grassland, that the object-based classifications classify about 10-15% more pixels as damage compared to the pixel-based classifications.

Table 9. The percentage-wise counts of pixels classified as Damage (i.e. Total of the Dry Damage and New Damage target classes) relative to the total amount of classified pixels for the grassland fields using the object and pixel-based RF and SVM classifications.

Grassland Percentage counts Damage	**Object-based RF**	**Object-Based SVM**	**Pixel-Based RF**	**Pixel-Based SVM**
Damage Pixels	47.36%	48.02%	37.41%	33.31%

5.1.3.1.4 Grassland: Difference Maps (2D-methods)

To further investigate the difference between the object and pixel-based classifications for grassland, difference maps were created (Figure S6 in Appendix 9.2 shows the difference maps of grassland field test 4 and wheatland field train 3). Refer to Table 10 for the frequency table created from the difference map of the grassland fields. The agreement was high between the pixel and object-based classifications for the different classifiers, most disagreement was at locations where the object-based classification classifies damage and pixel-based does not.

Table 10. Percentage-wise visualization of the averaged frequency tables of the difference maps of the pixel and object-based classifications for the Grassland fields. The pixels classified as Shadow were not included in the analysis.

Difference freq. Table Grass	**Random Forest**	**Support Vector Machine**
Pixel-based: Damage **Object-Based: No-Damage**	3.44%	2.80%
Agreement **(both damage/ no damage)**	82.71%	78.76%
Object-based: Damage **Pixel-Based: No-Damage**	13.85%	18.44%

5.1.3.2 Wheatland: Object & Pixel-based Classifications Test Results

5.1.3.2.1 Wheatland: Confusion Matrices and Test Evaluation Metrics

Table 11 shows that all the classifications had instances confusing True Dark Damage as Grass or Wheat. Next to this was True Wheat confused as Dry Damage. This is both ways around for pixel-based classifications. Table 12 shows that, as observed for the grassland, the object-based RF classification had lower Accuracy and Kappa values compared to the others. The F1 score is for all the classifications more or less similar, however, the object-based classifications had higher standard deviation scores for the F1 compared to the pixel-based

classifications. It can also be observed that for all the classifiers, the Producer's accuracy turned out lower, about 10%-15%, compared to their user's accuracy (this was not the case for the Object-Based SVM, here only a 4% difference). Lastly, the User's accuracies for the pixel-based classifications were around 95%, which is 5% higher compared to the object-based RF and SVM classifications.

Table 11. Confusion matrices containing the average values of the confusion matrices obtained from the test data origination from the separate fields used in the evaluation of the RF and SVM object-based and Pixel-based classifications for damage mapping in wheatland. The numbers represent pixels.

Object-Based RF	**True Dry Damage**	**True Grass**	**True Dark Damage**	**True Wheat**
Pred Dry Damage	**6001**	63	114	674
Pred Grass	77	**5566**	1294	106
Pred Dark Damage	101	270	**7648**	448
Pred Wheat	115	55	2127	**12033**

Object-Based SVM	**True Dry Damage**	**True Grass**	**True Dark Damage**	**True Wheat**
Pred Dry Damage	**6116**	64	129	1141
Pred Grass	53	**5744**	862	182
Pred Dark Damage	78	134	**8781**	269
Pred Wheat	56	12	1411	**11665**

Pixel-Based RF	**True Dry Damage**	**True Grass**	**True Dark Damage**	**True Wheat**
Pred Dry Damage	**5941**	28	28	387
Pred Grass	3	**5798**	536	17
Pred Dark Damage	58	56	**8161**	95
Pred Wheat	287	55	2454	**12735**

Pixel-Based SVM	**True Dry Damage**	**True Grass**	**True Dark Damage**	**True Wheat**
Pred Dry Damage	**5911**	31	23	612
Pred Grass	17	**5801**	412	6
Pred Dark Damage	99	58	**8442**	94
Pred Wheat	262	46	2301	**12521**

Table 12. Evaluation metrics were obtained by the object and pixel-based RF and SVM classifications on wheatland. Accuracy, Kappa and F1 were based on the 4-class classification performances. The producer and user accuracies were based on the Dry Damage and Dark Damage taken together as one damage class. Std. shows the standard deviation of the evaluation metric.

Wheatland Evaluation metrics	**Object-based RF**	**Object-Based SVM**	**Pixel-Based RF**	**Pixel-Based SVM**
Accuracy	0.85	0.88	0.89	0.90
Accuracy Std.	0.03	0.05	0.04	0.04
Kappa	0.79	0.84	0.85	0.85
Kappa Std.	0.04	0.06	0.05	0.05
F1	0.90	0.89	0.90	0.90
F1 std.	0.08	0.11	0.04	0.04
Damage Producer's Acc.	0.78	0.85	0.81	0.82
Damage Prod. Acc. Std.	0.06	0.07	0.06	0.07
Damage User's Acc	0.89	0.89	0.96	0.94
Damage User Acc. Std.	0.06	0.08	0.04	0.07

5.1.3.2.2 Wheatland: Classification Outputs (2D-methods)

Figure 8 shows the classifications for object and pixel-based field *train3* (Figure 1) for the wheatland, this field had the median amount of agreed classifications for both the object and pixel-based classifications (see Table S14 in Appendix 9.2). Similar to grassland, the pixel-based classifications mapped similar damage locations but mapped the damages less dense with the pixel-based RF mapping the least amount of damages at the damage locations.

5.1.3.2.3 Wheatland: Damage Counts (2D-methods)

Table 13 shows for wheatland, that the SVM classifications classified relatively more pixels as damage. Object and pixel-based comparison shows that the object-based classifies more pixels as damage.

Table 13. The percentage-wise counts of pixels classified as Damage (i.e. Total of the Dry Damage and Dark Damage target classes) relative to the total amount of classified pixels for the wheatland fields used in the object and pixel-based RF and SVM classifications.

Wheatland Percentage counts Damage	**Object-based RF**	**Object-Based SVM**	**Pixel-Based RF**	**Pixel-Based SVM**
Damage Pixels	27.79%	35.39%	20.52%	29.20%

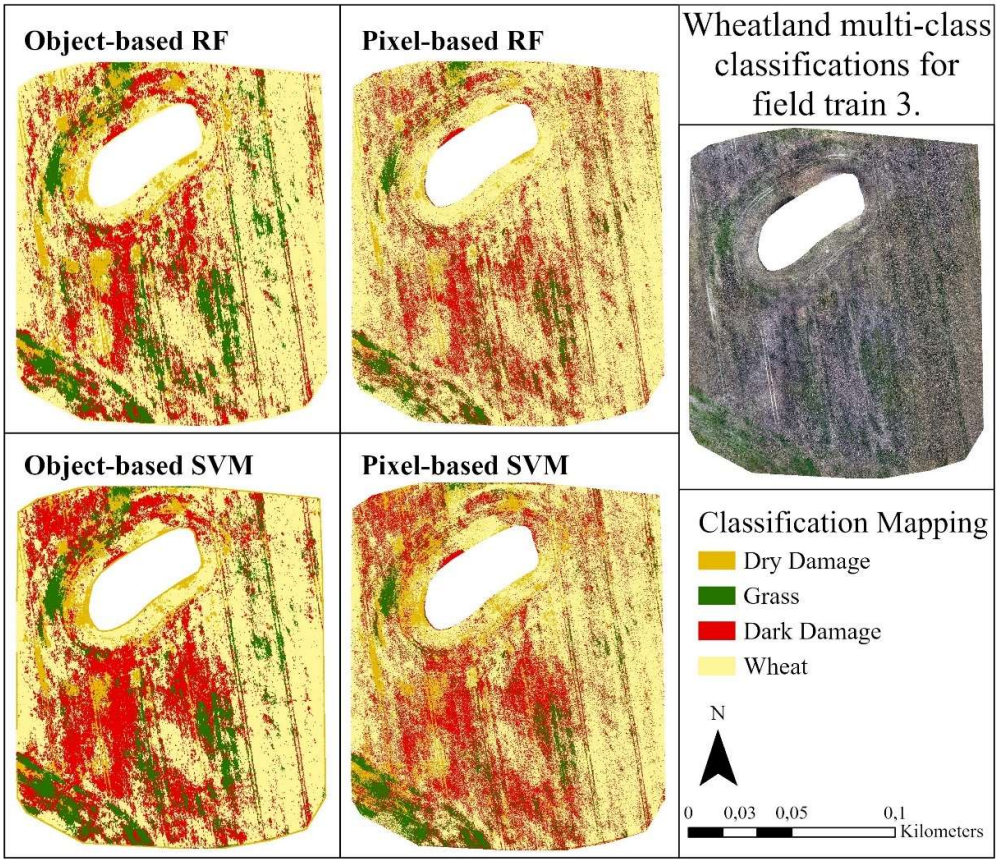

Figure 8. Visualization of the 4-class classifications of the object and pixel-based RF and SVM classifications for wheat field train 3 in coordinate reference system WGS 84 / UTM zone 33N. The data was collected in August 2023 in Boo, Hjortkvarn Municipality, Sweden using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters and a speed of 15m/s.

5.1.3.2.4 Wheatland: Difference Maps (2D-methods)

Refer to Figure S6 in Appendix 9.2 for the difference map of wheat field *train3*. Refer to Table 14 for the frequency tables created from the difference maps of the wheatland fields. The agreement is high between pixel and object-based classifications for the different classifiers, most disagreement is at locations where the object-based classification classifies damage and pixel-based does not.

Table 14. Percentage-wise visualization of the averaged frequency tables of the difference maps of the pixel and object-based classifications for the wheatland fields. The pixels classified as Grass were not included in the analysis.

Difference freq. Table Wheat	**Random Forest**	**Support Vector Machine**
Pixel-based: Damage Object-Based: No-Damage	6.11%	5.97%
Agreement (both damage/ no damage)	77.85%	79.85%
Object-based: Damage Pixel-Based: No-Damage	16.04%	14.18%

5.2 Point Cloud Height Threshold Classification Results (3D)

5.2.1 Point Cloud: Hyperparameter Results

Refer to S15 in Appendix 9.3 for the top 10 optimal hyperparameters found in the hyperparameter tuning of the point cloud classification method. It shows that the hyperparameters, classification threshold and the power, should be as low as possible and the number of neighbors as high as possible to obtain better classification performance. Table 15 shows the optimal hyperparameters.

Table 15. Results of the hyperparameter tuning of the point cloud height threshold classification method.

Classification Threshold	Power of proportional distance	Number of neighbors	Dice
1.00E-5	1	80	0.28

5.2.2 Point Cloud: Classification Outputs

The classification of wheat field *train 3* (Figure 9) shows that not all damages were captured. The damaged pits made by machinery were mapped as damage but the other damage spots, represented as gray areas in the RGB image, were less significantly detected, whereby only a few pixels were classified as Damage in these damage locations.

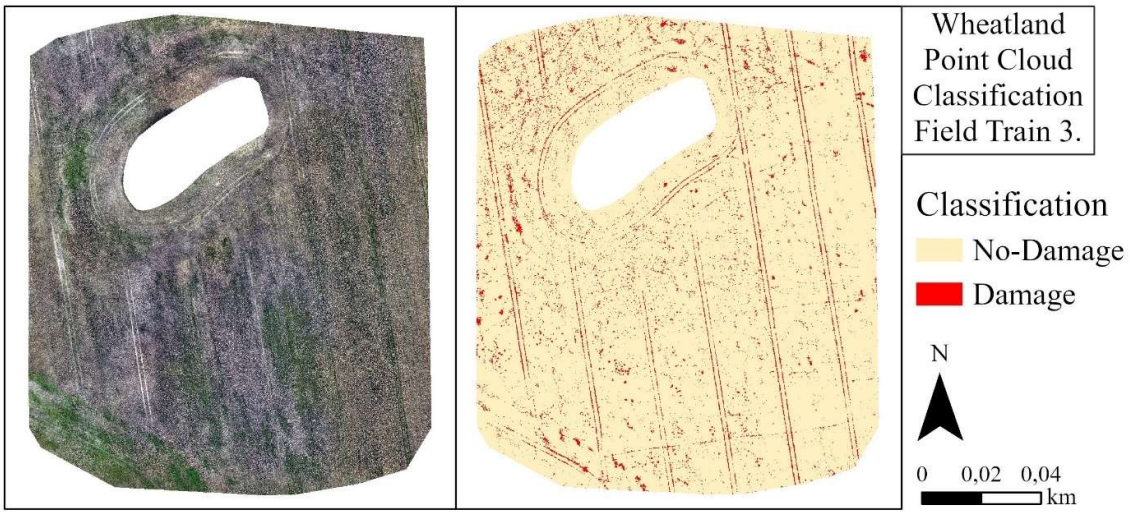

Figure 9. Resulting binary classification classified by the point cloud method of the wheat field train 3 in coordinate reference system WGS 84 / UTM zone 33N. The left shows the RGB image of the field and the right the binary classification. The data was collected in August 2023 in Boo, Hjortkvarn Municipality, Sweden using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters and a speed of 15m/s.

5.2.3 Point Cloud: Damage Counts

Percentage-wise counts of the damage relative to the total amount of classified pixels for the point cloud classification is 5.29%.

5.2.4. Point Cloud: Difference Maps 2D-3D

Figure S7 in Appendix 9.3 shows the difference maps between the wheat 2D object and pixel-based, RF and SVM classifications and the 3D point cloud classification. The difference maps show that the same regions were detected by the 2D and 3D classifications, but fewer pixels were classified as damage by the point cloud height threshold classification. Table 16 shows that most agreement was obtained by the pixel-based RF and the point cloud classification and the lowest agreement by the Object-based SVM. The disagreement was mostly due to the 2D classification classifying damage while the point cloud classification was not.

Table 16. Percentage-wise visualization of the averaged frequency tables of the difference maps of the wheat pixel and object-based classifications with the point cloud classification (i.e. 3D).

Difference freq. Table 2D – 3D Wheat	**Object-Based RF**	**Object-Based SVM**	**Pixel-Based RF**	**Pixel-Based SVM**
3D Damage, 2D No Damage	1.45%	1.08%	2.00%	1.49%
Agreement (both damage/ no damage)	70.64%	63.48%	78.76%	70.60%
2D Damage, 3D No Damage	27.91%	35.44%	19.24%	27.91%

5.3 Validation Field Data Comparison

5.3.1 Validation: Object and Pixel-based Classifications

Table 17 shows for both grass and wheatland, that the object-based RF and SVM detected a higher percentage of the validation field data compared to the pixel-based classifications (with the exception of wheat object-based RF). The standard deviations of the detections were of similar magnitude for the wheat fields for all classifications, for the grassland were the standard deviations higher for the pixel-based classification compared to object-based. Lastly, the grassland detections were higher compared to the wheatland detections of the validation data.

Table 17. Percentage-wise detection of the validation data for the grass and wheatland object and pixel-based RF and SVM classifications. The average and standard deviation of the fields' detections of the validation data are given in percentages.

Percentage Detection Validation data	**Object-based RF**	**Object-Based SVM**	**Pixel-Based RF**	**Pixel-Based SVM**
Grassland Average	94.32%	94.98%	89.93%	86.17%
Grassland Std.	4.23%	2.48%	9.77%	12.11%
Wheatland Average	64.44%	71.60%	53.86%	66.17%
Wheatland Std.	13.41%	12.21%	13.67%	14.42%

5.3.2 Validation: Point Cloud Height Threshold Classification

The average detection of the validation data in the different wheat fields was 19.33% with a standard deviation of 10.54%. Table 17 shows that relatively more of the validation field data was detected by the 2D methods compared to the 3D method for wheatlands.

6. Discussion

6.1 Key Findings

The object-based SVM performs best on the grassland damage mappings, as all the evaluation metrics (Table 8) obtained a score above 90% and as it obtained the highest detection of the validation field data (i.e. 95%, Table 17). The extra information that becomes available by the segmentation during the object-based classifications (i.e. mean and standard deviation of the bands of the grouped pixels) and the fact that the object-based SVM puts the most importance on the segment's standard deviations of multiple derived texture bands (i.e. RGB mean, NIR variance, and Nir mean, Table 5) resulted in a better performance of this classifier compared to the other 2D-based methods (i.e. object-based RF, pixel-based RF, and pixel-based SVM). These results were expected, as the UAV images for the grasslands did expect, that the texture variation between the different target classes would be distinct enough to be of use for the classification. The other classifiers, which put more importance on the spectral values themselves, also retrieved good mapping results. This was also expected, as it was observed that the spectral information had low heterogeneity within the target classes and had little overlap between the target classes. Lastly, it can be concluded that the collected UAV data (i.e. bands containing R, G. B, NIR, and RE sensory information) that was expected to be useful for the classifications, was of use for the classification, as each of these bands obtained high importance scores (Table 5) for all the classifiers (i.e. object and pixel-based RF and SVM).

The results of the 2D-based classifications for the wheatlands do not clearly show one classification method that performs best in damage mapping. However, depending on the purpose of use of the classifier, a distinction can be made as to what classifier performs best. First, when it is important that the overall damage mapping is as best as possible, expressed in the highest evaluation metrics for Accuracy, Kappa, and F1 (i.e. evaluation metrics based on the overall performance, subsection 4.3.2.5, Table 12) and the highest detection of the validation field data (Table 17), then the object-based SVM is best. This shows, similar to the results from grassland, that the information that becomes available due to the segments created in the object-based classification (i.e. mean and standard deviation of the bands of the grouped pixels) and the fact that the SVM classifier puts more importance on the derived texture bands and the standard deviations of the segments (Table 5), is improving the performance of the damage mapping in wheatlands. However, the pixel-based SVM can be concluded to perform best, when it is of importance that the classified damage areas are with high certainty true damage locations (e.g., low number of mistakes in classifying areas as damage that are not damage). This can be concluded from its high retrieved user's accuracy (i.e. 96% compared to 89% achieved by the object-based SVM, Table 12). However, the lower validation detection (i.e. 66% compared to 72% achieved by the object-based SVM, Table 17) and the lower producer's accuracy (i.e. 81% compared to 85% achieved by the object-based SVM, Table 12), show that this algorithm misses (greater than Object-based SVM) classifying all the damage locations and is classifying them as Wheat or Grass instead (Table 11). The finding that the object-based SVM performed overall best was as expected. Similar to grassland, the UAV images did expect that the texture variations between the

different target classes would be distinct enough to be useful for classification. However, dissimilar from grassland, it was observed that the spectral information of the pixels from the different target classes was very heterogeneous and contained overlapping values between the target classes. The fact that the other classifiers (i.e. object-based RF, pixel-based RF, and pixel-based SVM), which used mostly the spectral band information (e.g., R, G, B, RE, and NIR), performed worse, was therefore to be expected.

Lastly, it can be concluded that the standalone point cloud height threshold classification does not capture the damage mapping and that its performance is worse than the 2D classifications on wheatlands. The results from the point cloud classifications were not expected as the height differences between damaged areas and wheat were expected to differ enough to be able to map the damages. However, the resolution of the photogrammetry-derived point cloud caused the classification to miss and underrepresent the size of the damage locations.

The upcoming subsections contain a more detailed discussion of the different methods (i.e. Object-based, Pixel-Based, and Point Cloud Height Threshold Classification).

6.2 Object and Pixel-based Classifications (2D)

6.2.1 Object-based Classifications

6.2.1.1 Object-based: Segmentation Band

It was expected that the JM score's best band (i.e. RGB mean) would also be the band with the highest importance by the trained RF and SVM classifiers. As the JM score is calculated based on pixel-based values, it is difficult to relate the JM findings to the object-based band importances (Table 5), as these are based on the mean and standard deviation of the segments. However, it can be observed that the RGB mean is the most important band for the SVM-object-based classifications for both grass and wheatlands. The RF-object-based classifications deviate from the findings of the JM score as the RGB mean is the third and fifth most important band for grass and wheatland respectively. The JM scores can be related to the pixel-based results (Table 6) and as expected is the RGB mean band for each of the classifications (i.e. grass and wheatland, RF and SVM) the most important. These results show that the JM score is a good indicator of the band importances found by the different classifiers and has therefore the potential to be a valid method to be used to select the segmentation band. This is of importance, as this way of selecting a segmentation band saves a lot of computational time, as no separate hyperparameter tuning has to be performed to set this decision.

6.2.1.2 Object-based: Hyperparameter Tuning Results

The observation of the preference for a lower spatial range in Table S10 (Appendix 9.2) can be explained by the fact that an increasing search space (i.e. spatial range), increases the probability for higher heterogeneity within the pixel's spectral values of the created segments. This, in turn, results in an increased probability of single segments containing more than one target class. The preference for medium to high minimum sizes of the segments can be linked to the size of the polygons in the target class data. Given the landscape ecology of this study, the target class data used for training and testing contained damage spots bigger than 20 pixels (i.e. 0.05 m2). Therefore, very small segments (15 pixels) could lower the performance of the classification, as more confusion can occur when too many different segments represent one damage spot (explanation in section 4.3.2.3). It should therefore be noted that

when a study is also focusing on mapping very small damages (e.g. 0.01m2, 4 pixels), most probably lower minimum sizes would have been found optimal.

6.2.1.3 Object-based: Band Importances

The findings on the band importances are discussed in section 6.1. It can be concluded that different bands are important for the different machine learning classifiers RF and SVM. It cannot be expected that future studies using an object-based RF or SVM on damage mapping in agricultural grass and wheatlands will retrieve the same results. This is because the band importances are dependent on various factors such as the training data, the hyperparameters, and the randomness that is incorporated in the classifier algorithms, such as the random sample and feature selection within the RF classifier (subsection 4.3.2.2.1).

6.2.2 Pixel-based Classification

6.2.2.1 Pixel-based: Hyperparameter Tuning

The pixel-based hyperparameter tuning results of the RF (Figures S2:S5 in Appendix 9.2) show that a higher number of predictors (i.e. bands) lowered the performance. This can be explained by the fact that by having too many candidate predictors at the splits of the different decision trees, the trees have a higher probability of using similar bands to split on, thereby learning similar phenomena. This can lower the robustness of the trained RF with a lower performance as a result. The lower performance when too few predictors were used, is because it increases the chance that the bands that hold the important information for a high-performing classification will not be selected to be used in the decision tree training, resulting in worse performances.

The SVM's preference for a higher cost can be explained as follows. A lower cost can prevent your model from being too much affected by noise in the data (e.g., preventing overfitting), this is done by a lower penalty of these samples when they are too close to the separation line between the different target classes. However, when the cost is too low, the influence of the non-noise points on the separation line is also lowered, thereby increasing the chance of a too-generalized model that leads to more misclassification. The result of having better performance using the L2 loss could be because it penalized larger errors more than L1 (i.e. linear loss), meaning that the L2 loss made the SVM more robust to outliers.

The importance of performing hyperparameter tuning can be shown by these results and has also been studied by others (Yu and Zu, 2020; Yang and Shami, 2020). They show that the right balance has to be found to make the methods robust and to prevent them from overfitting to the training data. However, it should be noted that these machine learning algorithms are of a great part black boxes (Lipton, 2018), meaning that it can only be reasoned about what might have made the algorithm behave in a certain way. It is therefore important to realize that these hyperparameters are optimal specifically to this specific multi-class classification and can therefore not be generalized directly towards other classifications.

6.2.2.2 Pixel-based: Band Importance

The results of the SVM band importance (Table 6) might indicate that one texture band could have contained enough texture information, next to the sensory spectral information, for the SVM classifier to perform the classification.

6.2.3 Object & Pixel-based Classifier Training and Test Results

6.2.3.1 Grasslands: Object and Pixel-based Results (2D-methods)

The similar train (Table S11 in subsection 9.2) and test (Table 8) evaluation results indicate that the 4 train fields used for training the object and pixel-based classifiers contained the full spectrum of each target class's spectral and textural information that were present in each field used for testing the classification methods on grasslands.

The confusion matrices in Table 7 show confusion between instances of Shadow and New Damage for all the object and pixel-based classifiers. This can be explained by the fact that the class New Damage, due to fresh and therefore wetter soil, and the class Shadow, due to sunlight blocking, contained pixels with similar spectral values (e.g., lower spectral values) for the original sensory bands (i.e. R, G, B, NIR and RE). This resulted in a more difficult discrimination. Another important aspect of the Shadow target class that could confuse the classifications, is that it contained the shadow versions of the other target classes (i.e. the target classes dry Damage, New Damage, and Grass that are in the shadow were part of the Shadow target class). Therefore, values from the Shadow target class that are not influenced by the sunlight blocking such as the derived vegetation index and the texture values, may have been similar to those of the other target classes. Another confusion existing for all classifiers, is the confusion where True New Damage is classified as Dry Damage. This could be explained by the fact that these New Damage spots may have had lighter parts within the damage spot (e.g., spots with higher spectral values due to direct sunlight or dryness of the soil) resulting in overlapping spectral values with the Dry Damage target class, which confused the classifier. The Object-Based RF classification confuses True Dry Damage with New Damage. Why this is the case cannot be inferred given the information available on the classification.

Comparing the results of the object and pixel-based classifications (Table 7) shows that the object-based classification had more confusion between the Grass and New Damage classes. This is most probably the result of the segmentation of the object-based classifications that was done using only the RGB mean band. As Figure 3 shows, this band had a lot of overlapping values between these two target classes, meaning that there was a high chance that segments were created that consisted of both Grass and New Damage pixels. The misclassification could also be the result of the lower spectral distinction between the target classes, which is the result of using the mean and the standard deviation of the pixels in the segments for training the classifications (e.g., the individual pixel information is lost).

The standard deviations of the evaluation metrics, an indication of how stable the classifier is in performance for the different tested fields, show overall similar values for all the classifiers (Table 8). However, one deviation exists where the standard deviation of the F1 score was higher for the object-based classifications compared to the pixel-based classifications. This indicates that even though for all the fields there was a similar overall performance (i.e. performance measured on all target classes at once, accuracy and kappa), some fields have obtained a lot of misclassifications for some specific target classes while performing well on others (see section 4.3.2.5 for F1 explanation). This results in F1 scores that deviate largely between the fields, thereby receiving a higher standard deviation on this evaluation metric. Lastly should be noted that the highest mean evaluation metrics and the lowest standard deviations are retrieved by the Pixel-based SVM classifier, making it the most stable and best-performing classifier for damage mapping in grassland according to these results only.

The damage counts (Table 9), object and pixel-based agreements (Table 10), and the less dense classification of the damage locations by the pixel-based classification (Figure 7) can be explained by the fact that single pixels instead of segments were classified by the pixel-based classifications. This resulted in a more speckled (i.e., salt and pepper texture) classification, thereby classifying fewer pixels at those locations as damage.

6.2.3.2 Wheatlands: Object and Pixel-based Results (2D-methods)

The lower test evaluation results (Table 12) compared to the train results (Table S12) indicate that the 4 train fields used for training the object and pixel-based classifiers did not contain the full spectrum of each target class's spectral and textural information that was present in each field used for testing the classification methods on wheatlands. This was to be expected, as the wheatland data is a lot more heterogeneous in spectral and textural information, therefore a higher chance existed that the 4 train fields did not contain the full spectrum of each target class.

The confusion of all classifiers of classifying True Dark Damage as Wheat (Table 11) can be explained by the fact that these target classes had a wide range of spectral values that overlapped. The fact that less confusion existed for the object-based classifications is because they made use of the mean spectral values of the segments. Therefore these within-class differences were better filtered out, making the classes more homogenously represented to the classifier, thereby improving its performance. The greater confusion of the object-based classifiers where True Dark Damage was classified as Grass can again be explained due to the segmentation that was performed by one band. The RGB mean band used for the segmentations had a lot of overlapping values between these two target classes (Figure 4), resulting in segments that contained a mixture of these classes, thereby worsening the classification results. Lastly, the instances where all classifiers confuse True Wheat as Dry Damage could be explained by the fact that certain spots of the Wheat target class could have had high spectral values, as these spots' orientation was towards the sun, making them, as the Dry Damage, had high spectral values that overlap.

The result of the lower producer's accuracy (Table 12) compared to the user's accuracy, most significant for the pixel-based classifications, indicates that most of the damage classifications that were made are correct. However, the lower producer's accuracy indicates that damage spots were missed and classified as something else (i.e. Grass or Wheat).

The damage counts (Table 13), agreements (Table 14) and the observed damage mappings pattern (Figure 8) can be explained by the same reason as for grassland (section 6.2.3.1).

6.3 Point Cloud Height Threshold Classification (3D)

6.3.1 Point Cloud: Hyperparameter Tuning Results

The hyperparameter tuning results (Table S15 in Appendix 9.3) can be explained by the set objective for the DEM creation (i.e. the damage pits have a negative normalized height after the subtraction of the DEM). Using these hyperparameters, only a few points in the point cloud were selected as ground points and any points selected as ground points that might belong to a damage pit would be for a large part filtered out when creating the DEM. It should be noted that this way of DEM creation only works when working with terrain that does not contain steep slopes, which is usually the case for agricultural grounds.

6.3.2 Point Cloud: Test Evaluation Results

The low damage count (i.e. 5%) and the visualization of the output in Figure 9 show that not all damage was captured by the point cloud classification. This is because the too-low density point cloud filtered out small elevation differences in the fields that were needed by the method to map damages. The highest and lowest agreement with the pixel-based RF and Object-based SVM can be linked to the fact that these classifications also had, respectively, the lowest and highest damage counts (Table 13).

6.4 Validation Field Data Comparison

The low standard deviations (Table 17) of validation data detection results obtained by the object-based classifications for grassland, indicate that similar results were retrieved for each of the fields used in testing. The pixel-based detection of the validation field data was lower and had higher standard deviations meaning that less stable results were retrieved between the different grass fields used for testing. The results for the wheatland classifications show a lot lower detection and higher standard deviations compared to the grassland, indicating a worse damage mapping performance. Overall, it could be concluded that the SVM performs better than RF for the wheatlands. The low detection of validation data by the point cloud height threshold classification (i.e. 19%), resulted from the limited number of pixels classified as damage, attributed to the low-density point cloud used for this method.

6.5 Comparison With Other Studies

The study by Rutten et al. (2018) used an object-based RF to map damages in cornfields (vertical-growing crop comparable to wheat) and grasslands. They retrieved similar results, in that the classification of the grasslands was better than the vertical-growing cornfields. Rutten et al. (2018) decided to use several years of data on the cornfields to improve its classification accuracy. This master's thesis study observed that the trained classifiers for the wheatlands had not been able to learn the full scope of the damage mapping (subsection 6.2.3.2) by using the one-year available data for this study. This observation can be related to the study by Rutten et al. (2018) and therefore be an indication that multiple-year data will have the potential to improve the wheatland classifications. Rutten et al. (2018) trained the RF with a total of 25 bands among 8 textures and 9 spectra. Their band importances showed that the RF, similar to this study, put the most importance on the spectral values of the bands. A study done by Samiappan et al. (2018) used an object-based SVM to map damages in cornfields. They found that the classification was more likely to miss mapping damaged areas rather than mapping wrongly non-damaged areas as damage. Both the object and pixel-based SVM in this master's thesis study showed similar results, as the producer's accuracy is lower compared to the user's accuracy (Table 12). The accuracies found by the study by Samiappan et al. (2018) were lower compared to this study's object-based SVM wheat classification results (70% vs. 88% for the accuracy and 0.38 vs 0.84 for kappa). An explanation for this difference could be that Samiappan et al. (2018) had only an RGB sensor available, thereby missing information in the NIR and RE spectra. This spectral information helps the discrimination between soil and crop (elaborate explanation in subsection 4.3.1). Their absence has probably lowered their classification results.

A meta-analysis done on the RF and SVM classifiers for remote sensing image classification (Sheykhmousa et al.,2020) concluded that although the RF and SVM classifiers vary in performance per application, a superiority of one or the other can often be assigned based on the application (e.g., a RF is found superior to SVM to perform a land use and land cover

(LULC) mapping). Within the specific application of damage mapping in agricultural fields on a very-high landscape scale (i.e. cm-level), a superior is not yet established. This is most probably because UAV data is a relatively new form of data used for this kind of application, resulting in the low existence of studies that used the RF and SVM for damage mapping in agricultural fields. It is also important to notice the high diversity of decisions that can be made within this application. These include decisions on the different types of UAVs used for data capturing, the type of sensors with differences in capabilities and performances, spatial resolution, 2D/3D mapping methods, and the input data (e.g., quality of the captured UAV images, object vs pixel-based, and decisions on the spectra, derived textures, shapes, and vegetation indices to use). The damage mapping application can therefore be seen as a diverse and complex problem, thereby complicating the finding of one classifier superior to the other. One finding of the meta-analysis that can be linked to the better performance found of the SVM in this study, is that the SVM method is better at classifying data containing more features (i.e. the different bands in the ortho-mosaic) (Sheykhmousa et al.,2020). As UAV provides the ability to collect multispectral data and allows deriving additional data such as vegetation indices, textures, and shapes, a high amount of features will likely be used in this application to map damages in agricultural grounds. However, before any superiority can be concluded (if even possible), more studies on this topic are needed.

The study by Duro et al. (2012) focused on the statistical comparison of pixel-based with object-based classification on remote sensing image analysis of agricultural fields using medium spatial resolution satellite data. They similarly found no significant preference but found that the object-based classification gave a visually more appealing classification result. Yan et al. (2006) and Whiteside et al. (2011) found the object-based classification to perform significantly better compared to the pixel-based for a medium spatial resolution (e.g., several meters) remote sensing application. A possible explanation for why this finding was not observed in this study, can be related to the high spatial scale of this study. The behavior of the classifiers is dependent on the landscape factor of the input data. The high spatial resolution (i.e. cm-level) in this study resulted in higher heterogeneous input data compared to the data from the medium-scale landscapes used in the studies by Yan et al. (2006) and Whiteside et al. (2011). Thereby, the chance increased for the object-based classification to have highly heterogeneous segments that resulted in lower performances. This resulted in no significantly superior performance of the object-based classifications over the pixel-based classifications.

A similar method to the 3D method created in this study was studied by Michez et al. (2016). They performed a similar height threshold method on a photogrammetric-derived UAV point cloud to map cornfield damages. However, they retrieved more accurate results compared to this study. This difference in findings can be related to the fact that Michez et al. (2016) used a separate LiDAR-derived, high-quality digital terrain model to retrieve the normalized height values. Although their mapping reflected the validation field data better, they still had, similar to this study, an underestimation of the mapped damages. They linked their underestimation to flat-lying plant stems due to strong wind on the day of data capturing and that the damage areas were often too small to be represented well in their photogrammetric-derived UAV point clouds. In addition to these reasons, it is challenging to find a single height threshold that works in all situations as the heights of crops can differ between and within fields. This is due to external factors out of the operator's control, such as the acquisition date (e.g., reflecting the stage of the plant growth), weather conditions, and the health of the crops. However, it can be expected that higher-quality UAV point cloud data (e.g., LiDAR data, explanation in subsection 4.1) will be able to grasp the small differences

between damaged and non-damaged areas, even for small patches of damage, thereby potentially improving damage mappings in agricultural fields done with UAV 3D data.

With this study, a step is set towards a more automated and open-source method for UAV damage mapping in agricultural grass and wheatlands that the scientific community can build upon. The methods to map damages in agricultural wheat and grasslands developed in this study are performing well on grasslands and have a large potential to improve in performance for wheatland with higher data availability. The fact that these methods are almost fully automated and open source, improves the usability of this UAV-based damage mapping for real-life applications such as damage compensation to farmers and HWC management strategies risk analyses. The high agreement achieved by the damage mappings from the pixel-based and object-based methods shows the potential of the pixel-based method which is computationally less demanding. This study's findings thereby increase the potential of using UAV data for future real-life agricultural damage mapping, as the developed methods show that specific prior knowledge and high computational resources, often unavailable to users, become less of a necessity. However, the following limitations of this study should be considered.

6.6 Limitations

First, artifacts in the used UAV ortho-mosaics might have negatively influenced the results and the generalization of the developed methods (i.e. object-based, pixel-based, and point cloud height threshold classification). These artifacts can evolve during the data collection. Factors such as the roll, pitch, and yaw of the UAV during flight and other factors related to weather conditions (e.g., strong winds) can create errors in the data (e.g., motion blur, ghosting effects, and scale variations). Errors can arise as well within the mosaic creation software, a common procedure where errors arise is the tie point selection, which is used for stitching the UAV images into one ortho-mosaic. When working with highly heterogeneous data such as data from grass and especially wheatlands, this can be a challenging task as the unique features are limited (Samiappan et al., 2018).

Second, due to the time constraints and computational limits present in this study, not all available data was used for the analysis and the study area was narrowed down to one specific region in Sweden. As only a few fields of all the available fields were chosen to perform the analyses on, a part of the scope of the problem may be missed during the training of the classifications. This may have resulted in a lower generalization ability of the developed methods. Therefore, performance generalization of the methods to other locations in Sweden cannot be done without investigation. Also, not all hyperparameters present in the 2D and 3D methods were tuned using hyperparameter tuning and the hyperparameter values' search grids were set to smaller value ranges as would have been optimal. This might have resulted in the creation of methods that performed suboptimal.

Third, within this study, a high focus was on the creation of automated and open-source methods. Although the developed classification methods perform purely autonomously and are created with open-source programs, the program to process the raw UAV images into ortho-mosaics and point clouds (i.e. PIX4Dmapper version 4.8.4) is not. Another related limitation of the created methods is that the machine learning classifiers required human choices regarding what features (i.e. bands) would be relevant for the classifications. This means that the derived data (subsection 4.3.1), such as the texture values and the NDVI, are

hand-picked choices and created in the preprocessing of the input data, thereby reducing the objectivity and autonomy of the methods.

Fourth, the segmentation of the object-based classification method is computationally heavy and time-consuming, thereby lowering the method's usability in potential real-life applications.

Lastly, due to using a photogrammetric derived point cloud, the developed 3D method is not performing well on the damage mapping in wheatlands. However, the normalized height values that are automatically extracted by this method are expected to still have value (e.g., heights close to zero would potentially represent damages and larger heights undamaged crops) when used as an additional input band for the 2D-based methods. It should be noted that this has not been investigated.

6.7 Prospects

Prospects for this study can be related firstly to the existing confusion for all the classifiers between the Shadow and the New Damage target class for grasslands. It is suggested for future studies to split up the Shadow target class into different types of shadow versions of the other target classes. This will lower the heterogeneity within the Shadow target class and will thereby reduce the chance of confusion in the classifications. Another solution for this would be to only fly during cloudy conditions. Next to this, as discussed, some confusions at the object-based classifications arise due to segments containing a mix of target classes, a result of using one segmentation band only. Future studies investigating object-based classifications for this application should therefore use multiple segmentation bands to prevent heterogeneity of target classes within single segments.

Secondly, the agreements of about 80% between the object and pixel-based versions of the RF and SVM classifiers found in this study show the potential of the computationally lighter pixel-based method for damage mapping in agricultural grass and wheatlands. The pixel-based methods found the same damage locations but less dense (i.e. more speckled classification) than the object-based methods. Therefore it can be concluded that the potential exists for post-classification methods, such as the majority filter (i.e. filter that replaces the cell values of the classification output to the majority value of a specified number of continuous neighboring cells) (Townsend, 1986; Qian et al., 2005) to create denser classification maps out of the pixel-based results, thereby improving the damage mapping. To my knowledge, at this moment no studies utilizing UAV data to map damages in agricultural lands have investigated this computational lighter approach of pixel-based classification in detail. It is therefore suggested that this will be investigated in future studies.

Third, enhancing the automation of the developed methods. Work on deep learning approaches such as Convolutional Neural Networks (CNN) has shown their abilities to automatically learn relevant features directly from the original UAV ortho-mosaic (Szegedy et al., 2015). It is therefore recommended that future studies explore this technique as it has the potential to remove the human component from the UAV damage mapping, thereby improving the objectivity and future usability of the application.

The fourth prospect is related to the scale of the application of damage mapping in agricultural lands. Solely UAV data will not be enough to scale up the developed methods to the national level, which is often the scale at which HWC-related policy and management decisions must be made. Next to this, a need exists for consistent repetitive data collection

(e.g., annually) to create a valuable knowledge database on this topic and to enable the possibility of investigating the consequences of decisions. A possible solution to scale the application to the national level is by using airborne data. Such data is for example available via the Swedish Cadastral Agency (Lantmäteriet, National Land Survey of Sweden) which provides repetitive aerial photographs of the entire extent of Sweden, a product called the Digital Aerial photographs (Flygbilder Nedladdning). They collect aerial photographs in different combinations of wavelength bands, resolutions, and data collection repetitions. The higher densely populated regions in Sweden (i.e. the south) have a spatial resolution of 0.15m and are collected every third year. The less-dense regions, such as the east coast and the inland of Sweden are captured with a resolution of 0.37m every 4 and 6-10 years respectively. This data has the potential to be used for this application of damage mapping in agricultural lands due to its high spatial and temporal resolution. In 2021 they tested using as well a LIDAR during the aerial data collections. This can potentially be used for the 3D-based methods to map damages in agricultural lands.

Fifth, with the prospect of increasing the scale of the application, more efficient and less demanding methods are needed. Therefore, future research could focus on the enhancement of the overall computing performance of methods to map damages in agricultural lands. This can be achieved by techniques such as code optimization (e.g., parallelization and data structure optimization) and by high-performance computing (HPC) (i.e. aggregation of computer power to solve advanced computational problems).

Sixth, the automated damage mapping methods for agricultural grass and wheatland created in this study create a mapping containing both wild boar and other damages, such as machinery tracks, damages from other wildlife, and environmental-induced damages. Users such as farmers, policy and decision makers that want to use these damage maps for applications such as HWC management or economic loss compensation, have value by knowing, next to the exact damage locations, the cause of the damage. It is therefore of importance to investigate approaches that can automatically classify the damages into different types, such as wild boar, deer, batcher, insects, humans, drought and floods. Deep learning algorithms such as CNN could potentially be used to create a method that can automatically classify the types of damages. The performance of the CNN has been studied by Zhou et al. (2019) for classifying different crop types such as rice, peanuts, and corn in multi-spectral remote sensing data from Yuanyang county, China, and achieved an accuracy of 95%. The advantage of using this algorithm is that it uses feature maps that can automatically learn distinctive characteristics of each of the different damage types in the UAV images, thereby learning to classify them. However, it should be noted that this algorithm requires field data that contains the cause of damage, something that is often difficult to determine in the field post-time of the damage occurrence.

7. Conclusion

The aim of the research has been met by this study's development of two 2D-based UAV methods (i.e. object-based and pixel-based classification) and one 3D-based UAV method (i.e. point cloud height threshold classification) for automatic damage mapping in agricultural wheat and grasslands. This was achieved by their development in the open-source R-Studio version 4.4.0 (R core Team (2024)) in the form of R-scripts that run fully automatically from derived ortho-mosaics and point clouds on, where required manual user input is automatized via hyperparameter tuning and the use of statistical methods such as the JM-score. High performances of all the 2D-based methods were achieved. The object and pixel-based RF and SVM classifications retrieved overall accuracies of 91% and above for the grasslands and 85% and above for the wheatland. The object-based SVM classification obtained the overall best performance for both the damage mapping in grass and wheatland, with an overall accuracy and average validation field data detection of 93% and 95% respectively for grassland, and 88% and 72% respectively for wheatland. The agreement of about 80% between the object and pixel-based RF and SVM classifications for both wheat and grassland showed that to a certain extent, similar damage mappings were retrieved from the pixel and object-based classifications. Their classifications showed that they mapped similar damage locations but that the pixel-based classifications underestimated the size of damage at those locations. The point cloud height threshold classification obtained low performances with an average validation field data detection of 19% for wheatlands. The method automatically derives the normalized heights from the photogrammetry-derived point clouds, thereby having the potential to aid the 2D-based classifications for mapping damages in agricultural wheatlands when used as an additional information band available to use by the classifiers.

8. References

Addink, E. A., De Jong, S., Davis, S., Dubyanskiy, V. M., Burdelov, L. A., & Leirs, H. (2010). The use of high-resolution remote sensing for plague surveillance in Kazakhstan. Remote Sensing of Environment, 114(3), 674–681. https://doi.org/10.1016/j.rse.2009.11.015

AIML.com. (2023). What are the advantages and disadvantages of Random Forest? *AIML.com*. https://aiml.com/what-are-the-advantages-and-disadvantages-of-random-forest/

Belgiu, M., & Drăguţ, L. (2016). Random forest in remote sensing: A review of applications and future directions. *ISPRS Journal of Photogrammetry and Remote Sensing*, *114*, 24–31. https://doi.org/10.1016/j.isprsjprs.2016.01.011

Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of machine learning research, 13(2).

Blaschke, T. (2010). Object based image analysis for remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, *65*(1), 2–16. https://doi.org/10.1016/j.isprsjprs.2009.06.004

Blaschke, T., Hay, G. J., Kelly, M., Lang, S., Hofmann, P., Addink, E. A., Feitosa, R. Q., Van Der Meer, F. D., Van Der Werff, H., Van Coillie, F., & Tiede, D. (2014). Geographic Object-Based Image Analysis – Towards a new paradigm. *ISPRS Journal of Photogrammetry and Remote Sensing*, *87*, 180–191. https://doi.org/10.1016/j.isprsjprs.2013.09.014

Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In Proceedings of the fifth annual workshop on Computational learning theory (pp. 144-152). https://doi.org/10.1145/130385.130401

Breiman, L. (2001). Random Forests. *Machine Learning*, *45*(1), 5–32. https://doi.org/10.1023/a:1010933404324

Bruzzone, L., Roli, F., & Serpico, S. B. (1995). An extension of the Jeffreys-Matusita distance to multiclass cases for feature selection. *IEEE Transactions on Geoscience and Remote Sensing*, *33*(6), 1318–1321. https://doi.org/10.1109/36.477187

Chapelle, O., Vapnik, V., Bousquet, O., & Mukherjee, S. (2002). Choosing Multiple Parameters for Support Vector Machines. *Machine Learning*, *46*(1/3), 131–159. https://doi.org/10.1023/a:1012450327387

Dabboor, M., Howell, S., Shokr, M., & Yackel, J. J. (2014). The Jeffries–Matusita distance for the case of complex Wishart distribution as a separability criterion for fully polarimetric SAR data. *Journal of Remote Sensing*, *35*(19), 6859–6873. https://doi.org/10.1080/01431161.2014.960614

Dabrowska-Zielinska, K., Budzynska, M., Malek, I., Bojanowski, J., Bochenek, Z., & Lewinski, S. (2009). Assessment of crop growth conditions for agri–environment ecosystem for modern landscape management. In Remote Sensing for a Changing Europe (pp. 247-254). IOS Press. https://doi.org/10.3233/978-1-58603-986-8-247

Dalponte, M., Ørka, H. O., Gobakken, T., Gianelle, D., & Næsset, E. (2013). Tree species classification in boreal forests with hyperspectral data. *IEEE Transactions on Geoscience and Remote Sensing*, *51*(5), 2632–2645. https://doi.org/10.1109/tgrs.2012.2216272

Dalponte, M., Ørka, H. O., (2021*). _varSel: Sequential Forward Floating Selection using Jeffries-Matusita Distance [R package version 0.2*]. https://CRAN.R-project.org/package=varSel

Dickman, A. (2010). Complexities of conflict: the importance of considering social factors for effectively resolving human–wildlife conflict. *Animal Conservation*, *13*(5), 458–466. https://doi.org/10.1111/j.1469-1795.2010.00368.x

Ding, N., Atzeni, L., Chen, Y., Lyu, Z., & Shi, K. (2023). Mapping crop damage by wild boars using multi-scale risk modeling in Northeast China. *The Journal of Wildlife Management*, *87*(6), e22418.https://doi.org/10.1002/jwmg.22418

DJI. (n.d.). *Mavic 3 multispectral edition - see more, work smarter – DJI agricultural drones*. DJI Agriculture. https://ag.dji.com/mavic-3-m

Duro, D. C., Franklin, S. E., & Dubé, M. G. (2012). A comparison of pixel-based and object-based image analysis with selected machine learning algorithms for the classification of agricultural landscapes using SPOT-5 HRG imagery. *Remote Sensing of Environment*, *118*, 259–272. https://doi.org/10.1016/j.rse.2011.11.020

Edman, M., & Ländell, G. (2021). *Damage to agricultural crops caused by wildlife has doubled*. Statistics Sweden. https://www.scb.se/en/finding-statistics/statistics-by-subject-area/agriculture-forestry-and-fishery/agricultural-production/production-of-cereals-dried-pulses-and-oil-seeds/pong/statistical-news/production-of-cereals-dried-pulses-and-oil-seeds-2020/

Ensrud, K. E. (2021). Epidemiologic methods in studies of osteoporosis. In *Elsevier eBooks* (pp. 381–403). https://doi.org/10.1016/b978-0-12-813073-5.00017-4

Eskandari, R., Mahdianpari, M., Mohammadimanesh, F., Salehi, B., Brisco, B., & Homayouni, S. (2020). Meta-analysis of unmanned aerial vehicle (UAV) imagery for agro-environmental monitoring using machine learning and statistical models. *Remote Sensing*, *12*(21), 3511. https://doi.org/10.3390/rs12213511

Fall, M. W., & Jackson, W. B. (2002). The tools and techniques of wildlife damage management—changing needs: an introduction. *International Biodeterioration & Biodegradation*, *49*(2–3), 87–91. https://doi.org/10.1016/s0964-8305(01)00107-x

Feng, Q., Liu, J., & Gong, J. (2015). UAV remote sensing for urban vegetation mapping using random forest and texture analysis. *Remote Sensing*, *7*(1), 1074–1094. https://doi.org/10.3390/rs70101074

Fischer, J. W., Greiner, K., Lutman, M. W., Webber, B. L., & VerCauteren, K. C. (2019). Use of unmanned aircraft systems (UAS) and multispectral imagery for quantifying agricultural areas damaged by wild pigs. *Crop Protection*, *125*, 104865. https://doi.org/10.1016/j.cropro.2019.104865

Friesenhahn, B. A., Massey, L. D., DeYoung, R. W., Cherry, M. J., Fischer, J. W., Snow, N. P., VerCauteren, K. C., & Perotto-Baldivieso, H. L. (2023). Using drones to detect and quantify wild pig damage and yield loss in corn fields throughout plant growth stages. *Wildlife Society Bulletin*, *47*(2). https://doi.org/10.1002/wsb.1437

Ge, X., Wang, J., Ding, J., Cao, X., Zhang, Z., Liu, J., & Li, X. (2019). Combining UAV-based hyperspectral imagery and machine learning algorithms for soil moisture content monitoring. *PeerJ*, *7*, e6926. https://doi.org/10.7717/peerj.6926

Gonçalves, J., Pôças, I., Marcos, B., Mücher, C., & Honrado, J. P. (2019). SegOptim—A new R package for optimizing object-based image analyses of high-spatial resolution remotely-sensed data. *International Journal of Applied Earth Observation and Geoinformation*, *76*, 218–230. https://doi.org/10.1016/j.jag.2018.11.011

Gren, I., Andersson, H., Mensah, J., & Pettersson, T. (2019). Cost of wild boar to farmers in Sweden. *European Review of Agricultural Economics*. https://doi.org/10.1093/erae/jbz016

Haralick, R. M., Shanmugam, K., & Dinstein, I. (1973). Textural features for image classification. *IEEE Transactions on Systems, Man, and Cybernetics*, *SMC-3*(6), 610–621. https://doi.org/10.1109/tsmc.1973.4309314

Hay, G. J., & Castilla, G. (2008). Geographic Object-Based Image Analysis (GEOBIA): A new name for a new discipline. In *Lecture notes in geoinformation and cartography* (pp. 75–89). https://doi.org/10.1007/978-3-540-77058-9_4

Heremans, S., Dong, Q., Zhang, B., Bydekerke, L., & Van Orshoven, J. (2015). Potential of ensemble tree methods for early-season prediction of winter wheat yield from short time series of remotely sensed normalized difference vegetation index andin situmeteorological data. *Journal of Applied Remote Sensing*, *9*(1), 097095. https://doi.org/10.1117/1.jrs.9.097095

Ilich, Alexander R. 2020. “GLCMTextures”, doi:10.5281/zenodo.4310186. https://github.com/ailich/GLCMTextures

International Union for Conservation of Nature [IUCN]. (2022, June). *Human-Wildlife Conflict.* IUCN. https://www.iucn.org/sites/default/files/2022-06/iucn-issues-brief-human-wildlife-conflict_final.pdf

Jayathunga, S., Owari, T., & Tsuyuki, S. (2018). The use of fixed–wing UAV photogrammetry with LiDAR DTM to estimate merchantable volume and carbon stock in living biomass over a mixed conifer–broadleaf forest. *International Journal of Applied Earth Observation and Geoinformation*, *73*, 767–777. https://doi.org/10.1016/j.jag.2018.08.017

Jin, Y., Kong, W., Yan, H., Bao, G., Liu, T., Ma, Q., Li, X., Zou, H., & Zhang, M. (2021). Multi-Scale spatial prediction of wild boar damage risk in Hunchun: a key tiger range in China. *Animals*, *11*(4), 1012. https://doi.org/10.3390/ani11041012

Journel, A. G., & Huijbregts, C. J. (1997). Mining Geostatistics. *Mineralogical Magazine*, *43*(328), 563–564. https://doi.org/10.1180/minmag.1979.043.328.34

Kuhn, M. (2008). Building Predictive Models inRUsing theCaretPackage. *Journal of Statistical Software*, *28*(5). https://doi.org/10.18637/jss.v028.i05

Kumar, M., Bhattacharya, B. K., Pandya, M. R., & Handique, B. K. (2024). Machine learning based plot level rice lodging assessment using multi-spectral UAV remote sensing. *Computers and Electronics in Agriculture*, *219*, 108754. https://doi.org/10.1016/j.compag.2024.108754

Kuželka, K., & Surový, P. (2018). Automatic detection and quantification of wild game crop damage using an unmanned aerial vehicle (UAV) equipped with an optical sensor payload: a case study in wheat. *European Journal of Remote Sensing*, *51*(1), 241–250. https://doi.org/10.1080/22797254.2017.1419442

Hubing, N. (2023, August 2). *Satellite Imagery Pricing - Home - Aerial/Satellite Digital Mapping Solutions - LAND INFO . . . landinfo.com*. LAND INFO Worldwide Mapping. https://landinfo.com/satellite-imagery-pricing/Lemenkova, P. (2021). ISO Cluster classifier by ArcGIS for unsupervised classification of the Landsat TM image of Reykjavík. Bulletin of Natural Sciences Research, 11(1), 29-37.

Liaw, A., & Wiener, M. (2002). Classification and regression by randomForest. R news, 2(3), 18-22.

Lin, H., Żebrowski, P., Fath, B. D., Liljenström, H., & Rovenskaya, E. (2021). Modelling stakeholder satisfaction for conflict resolution in wildlife management: a case of wolf population in Sweden. *European Journal of Wildlife Research*, *67*(4). https://doi.org/10.1007/s10344-021-01495-1

Lipton, Z. C. (2018). The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, *16*(3), 31-57.

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C. C., & Lin, C. C. (2019). e1071: misc functions of the department of statistics, probability theory group (formerly: E1071), TU Wien. R package version, 1(2).

Michel, J., Youssefi, D., & Grizonnet, M. (2015). Stable Mean-Shift algorithm and its application to the segmentation of arbitrarily large remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, *53*(2), 952–964. https://doi.org/10.1109/tgrs.2014.2330857

Michez, A., Morelle, K., Lehaire, F., Widar, J., Authelet, M., Vermeulen, C., & Lejeune, P. (2016). Use of unmanned aerial system to assess wildlife (Sus scrofa) damage to crops (Zea mays). *Journal of Unmanned Vehicle Systems*, *4*(4), 266–275. https://doi.org/10.1139/juvs-2016-0014

Miller, J. R. B. (2015). Mapping attack hotspots to mitigate human–carnivore conflict: approaches and applications of spatial predation risk modeling. *Biodiversity and Conservation*, *24*(12), 2887–2911. https://doi.org/10.1007/s10531-015-0993-6

Mongus, D., & Žalik, B. (2012). Parameter-free ground filtering of LiDAR data for automatic DTM generation. *ISPRS Journal of Photogrammetry and Remote Sensing*, *67*, 1–12. https://doi.org/10.1016/j.isprsjprs.2011.10.002

Mulla, D. J. (2013). Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. *Biosystems Engineering*, *114*(4), 358–371. https://doi.org/10.1016/j.biosystemseng.2012.08.009

Müllerová, J., Brůna, J., Bartaloš, T., Dvořák, P., Vítková, M., & Pyšek, P. (2017). Timing Is Important: Unmanned Aircraft vs. Satellite Imagery in Plant Invasion Monitoring. *Frontiers in Plant Science*, *8*. https://doi.org/10.3389/fpls.2017.00887

Najah, A., El-Shafie, A., Karim, O. A., Jaafar, O., & El-Shafie, A. H. (2011). An application of different artificial intelligences techniques for water quality prediction. *International Journal of the Physical Sciences*, *6*(22), 5298–5308. https://doi.org/10.5897/ijps11.1180

Nguyen, T. T., Hoang, T. D., Pham, M. T., Vu, T. T., Nguyen, T. H., Huynh, Q., & Jo, J. (2020). Monitoring agriculture areas with satellite images and deep learning. *Applied Soft Computing*, *95*, 106565. https://doi.org/10.1016/j.asoc.2020.106565

Pádua, L., Vanko, J., Hruška, J., Adão, T., Sousa, J. J., Peres, E., & Morais, R. (2017). UAS, sensors, and data processing in agroforestry: a review towards practical applications. *International Journal of Remote Sensing*, *38*(8–10), 2349–2391. https://doi.org/10.1080/01431161.2017.1297548

Palaniswami, C., Upadhyay, A. K., & Maheswarappa, H. P. (2006). Spectral mixture analysis for subpixel classification of coconut. *Current Science Association*, *91*(12), 1706–1711. https://www.jstor.org/stable/24094026

Perumal, K., & Bhaskaran, R. (2010). Supervised classification performance of multispectral images. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.1002.4046

Peterson, M. N., Birckhead, J. L., Leong, K., Peterson, M. J., & Peterson, T. R. (2010). Rearticulating the myth of human–wildlife conflict. *Conservation Letters*, *3*(2), 74–82. https://doi.org/10.1111/j.1755-263x.2010.00099.x

Qian, Y., Zhang, K., & Qiu, F. (2005). Spatial contextual noise removal for post classification smoothing of remotely sensed images. *SAC '05: Proceedings of the 2005 ACM Symposium on Applied Computing*, 524–528. https://doi.org/10.1145/1066677.1066795

R Core Team (2023). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rao, K., Maikhuri, R., Nautiyal, S., & Saxena, K. (2002). Crop damage and livestock depredation by wildlife: a case study from Nanda Devi Biosphere Reserve, India. *Journal of Environmental Management*, *66*(3), 317–327. https://doi.org/10.1006/jema.2002.0587

Ravenelle, J., & Nyhus, P. J. (2017). Global patterns and trends in human–wildlife conflict compensation. *Conservation Biology*, *31*(6), 1247–1256. https://doi.org/10.1111/cobi.12948

Rocchini, D. (2007). Effects of spatial and spectral resolution in estimating ecosystem α-diversity by satellite imagery. *Remote Sensing of Environment*, *111*(4), 423–434. https://doi.org/10.1016/j.rse.2007.03.018

Rossler, S. T., Gehring, T. M., Schultz, R. N., Rossler, M. T., Wydeven, A. P., & Hawley, J. E. (2012). Shock collars as a site-aversive conditioning tool for wolves. *Wildlife Society Bulletin*, *36*(1), 176–184. https://doi.org/10.1002/wsb.93

Rostro-García, S., Tharchen, L., Abade, L., Astaras, C., Cushman, S. A., & Macdonald, D. W. (2016). Scale dependence of felid predation risk: identifying predictors of livestock kills by tiger and leopard in Bhutan. *Landscape Ecology*, *31*(6), 1277–1298. https://doi.org/10.1007/s10980-015-0335-9

Rouse, J. W., Haas, R. H., Schell, J. A., & Deering, D. W. (1974). Monitoring Vegetation Systems in the Great Plains with Erts. *NASA Special Publication*, *351*, 309. http://ui.adsabs.harvard.edu/abs/1974NASSP.351..309R/abstract

Roussel, J. R., Auty, D., Coops, N. C., Tompalski, P., Goodbody, T. R., Meador, A. S., Bourdon, J. F., De Boissieu, F., & Achim, A. (2020). lidR: An R package for analysis of Airborne Laser Scanning (ALS) data. *Remote Sensing of Environment*, *251*, 112061. https://doi.org/10.1016/j.rse.2020.112061

Roussel, J. R. & Auty, D. (2024). *Airborne LiDAR Data Manipulation and Visualization for Forestry Applications [R package version 4.1.1].* https://cran.r-project.org/package=lidR

Rusu, R. B., & Cousins, S. (2011). 3D is here: Point Cloud Library (PCL). *IEEE International Conference on Robotics and Automation*. https://doi.org/10.1109/icra.2011.5980567

Rutten, A., Casaer, J., Vogels, M. F. A., Addink, E. A., Vanden Borre, J., & Leirs, H. (2018). Assessing agricultural damage by wild boar using drones. *Wildlife Society Bulletin*, *42*(4), 568–576. https://doi.org/10.1002/wsb.916

Samiappan, S., Czarnecki, J. M. P., Foster, H., Strickland, B. K., Tegt, J. L., & Moorhead, R. J. (2018). Quantifying damage from wild pigs with small unmanned aerial systems. *Wildlife Society Bulletin*, *42*(2), 304–309. https://doi.org/10.1002/wsb.868

Scholkopf, B., & Smola, A. J. (2018). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press.

Schut, G. H. (1976). Review of interpolation methods for digital Terrain models. *Canadian Surveyor*, *30*(5), 389–412. https://doi.org/10.1139/tcs-1976-0037

Sheykhmousa, M., Mahdianpari, M., Ghanbari, H., Mohammadimanesh, F., Ghamisi, P., & Homayouni, S. (2020). Support Vector Machine versus Random Forest for Remote Sensing Image Classification: A Meta-Analysis and Systematic Review. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, *13*, 6308–6325. https://doi.org/10.1109/jstars.2020.3026724

Sjoegren, A., & Matsuda, H. (2016). Seeing the Wolf through Sami Eyes – Understanding Human Dimensions of Wildlife Conflict in Northern Sweden. *International Journal Sustainable Future for Human Security*, *4*(1), 35–49.

Skakun, S., Kussul, N., Shelestov, A., & Kussul, O. (2015). The use of satellite data for agriculture drought risk quantification in Ukraine. *Geomatics, Natural Hazards & Risk*, *7*(3), 901–917. https://doi.org/10.1080/19475705.2015.1016555

Snoek, J., Larochelle, H., & Adams, R. P. (2012). Practical Bayesian Optimization of Machine Learning Algorithms. In F. Pereira, C. J. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in Neural Information Processing Systems* (Vol. 25). Retrieved from https://proceedings.neurips.cc/paper_files/paper/2012/file/05311655a15b75fab86956663e1819cd-Paper.pdf

Stas, M., Van Orshoven, J., Dong, Q., Heremans, S., & Zhang, B. (2016). A comparison of machine learning algorithms for regional wheat yield prediction using NDVI time series of SPOT-VGT. In *International Conference on Agro-geoinformatics (Agro-geoinformatics)* (5th ed., pp. 1–5). https://doi.org/10.1109/agro-geoinformatics.2016.7577625

Jordbruksverket. (2021). KVALITETSDEKLARATION Viltskador i lantbruksgrödor. In *scb.se* (No. JO0601). https://www.scb.se/contentassets/462f77b26e0d4febbbaab8b204d960f5/jo0601_kd_2020_gl_20210705.pdf

Su, T. (2017). A study of a matching pixel by pixel (MPP) algorithm to establish an empirical model of water quality mapping, as based on unmanned aerial vehicle (UAV) images. *International Journal of Applied Earth Observation and Geoinformation*, *58*, 213–224. https://doi.org/10.1016/j.jag.2017.02.011

Swedish Association for Hunting and Wildlife Management (2017) Game monitoring. Available at: https://jagareforbundet.se/vilt/viltovervakning/digital-viltovervakning/

Szegedy, C., Liu, N. W., Jia, N. Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., & Rabinovich, A. (2015). Going deeper with convolutions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1–9. https://doi.org/10.1109/cvpr.2015.7298594

Torres, D. F., Oliveira, E. S., & Alves, R. R. N. (2018). Conflicts between humans and terrestrial vertebrates: A global review. *Tropical Conservation Science*, *11*, 194008291879408. https://doi.org/10.1177/1940082918794084

Townsend, F. E. (1986). The enhancement of computer classifications by logical smoothing. *Photogrammetric Engineering and Remote Sensing*, *52*(2), 213-221.

Vapnik, V. N. (1998). *Statistical learning theory*. Wiley-Interscience.

Vapnik, V. (2013). *The nature of statistical learning theory*. Springer Science & Business Media.

Wang, S., Garcia, M., Ibrom, A., Jakobsen, J., Köppl, C. J., Mallick, K., Looms, M. C., & Bauer-Gottwein, P. (2018). Mapping Root-Zone Soil Moisture Using a Temperature–Vegetation Triangle Approach with an Unmanned Aerial System: Incorporating Surface Roughness from Structure from Motion. *Remote Sensing*, *10*(12), 1978. https://doi.org/10.3390/rs10121978

Whiteside, T. G., Boggs, G. S., & Maier, S. W. (2011). Comparing object-based and pixel-based classifications for mapping savannas. *International Journal of Applied Earth Observation and Geoinformation*, *13*(6), 884–893. https://doi.org/10.1016/j.jag.2011.06.008

Woodcock, C. E., & Strahler, A. H. (1987). The factor of scale in remote sensing. *Remote Sensing of Environment*, *21*(3), 311–332. https://doi.org/10.1016/0034-4257(87)90015-0

Vailshery, L. S. (2022, July 12). *Commercial satellite imagery cost worldwide 2022 | Statista*. Statista. https://www.statista.com/statistics/1293877/commercial-satellite-imagery-cost-worldwide/

Van Der Werff, H., & Van Der Meer, F. (2008). Shape-based classification of spectrally identical objects. *ISPRS Journal of Photogrammetry and Remote Sensing*, *63*(2), 251–258. https://doi.org/10.1016/j.isprsjprs.2007.09.007

Yang, L., & Shami, A. (2020). On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, *415*, 295–316. https://doi.org/10.1016/j.neucom.2020.07.061

Yu, T., & Zhu, H. (2020). Hyper-Parameter Optimization: A review of Algorithms and applications. *arXiv (Cornell University)*. https://doi.org/10.48550/arxiv.2003.05689

Zhang, W., Qi, J., Wan, P., Wang, H., Xie, D., Wang, X., & Yan, G. (2016). An Easy-to-Use airborne LiDAR data filtering method based on cloth simulation. *Remote Sensing*, *8*(6), 501. https://doi.org/10.3390/rs8060501

Zhou, Z., Li, S., Zhang, K., & Shao, Y. (2019). Crop mapping using remotely sensed spectral and context features based on CNN. *Remote Sensing Technology and Application*, *34*(4), 694-703.

9. Appendix

9.1 Data Collection

Table S1. The total amount of hectares the train and test fields consisted of. The individual values correspond to the areas of the separate fields present in the train and test datasets. The data was collected in May 2023 for grassland and in August 2024 for wheatland in Boo, Hjortkvarn Municipality, Sweden using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters and a speed of 15m/s.

What	Area (hectares)
Train grassland	2.2 (Tr1) + 4.0 (Tr2) + 3.2 (Tr 3) + 1.6 (Tr4) = 11
Test grassland	3.1 (Te1) + 1.0 (Te2) + 1.3 (Te3) + 3.1 (Te4) = 8.5
Train wheatland	1.8 (Tr1) + 2.4 (Tr2) + 2.6 (Tr3) + 4.1 (Tr4) = 10.9
Test wheatland	1.9 (Te1) + 2.4 (Te2) + 2.9 (Te3) + 2.4 (Te4) = 9.6

Table S2. The UAV ortho-mosaics (RGB display) in coordinate reference system WGS 84 / UTM zone 33N created from the drone images of the 4 train fields of grassland. The data was collected in May 2023 in Boo, Hjortkvarn Municipality, Sweden, using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters, and a speed of 15m/s.

			
Train 1	**Train 2**	**Train 3**	**Train 4**

Table S3. The UAV ortho-mosaics (RGB display) in coordinate reference system WGS 84 / UTM zone 33N created from the drone images of the 4 test fields of grassland. The data was collected in May 2023 in Boo, Hjortkvarn Municipality, Sweden, using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters, and a speed of 15m/s.

			
Test 1	**Test 2**	**Test 3**	**Test 4**

Table S4. The UAV ortho-mosaics (RGB display) in coordinate reference system WGS 84 / UTM zone 33N created from the drone images of the 4 train fields of wheatlands. The data was collected in August 2023 in Boo, Hjortkvarn Municipality, Sweden, using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters, and a speed of 15m/s.

			
Train 1	**Train 2**	**Train 3**	**Train 4**

Table S5. The UAV ortho-mosaics (RGB display) in coordinate reference system WGS 84 / UTM zone 33N created from the drone images of the 4 test fields of wheatland. The data was collected in August 2023 in Boo, Hjortkvarn Municipality, Sweden, using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters, and a speed of 15m/s.

	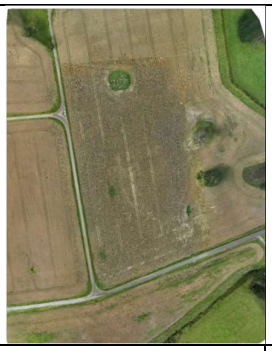		
Test 1	**Test 2**	**Test 4**	**Test 5**

Table S6. Quality report numbers for each of the selected grass and wheat training fields separately that were retrieved by processing the raw UAV images in PIX4Dmapper version 4.8.4 with a pixel size of 5cm. The UAV images were collected in May 2023 for grassland and August 2023 for wheatland in Boo, Hjortkvarn Municipality, Sweden, using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters, and a speed of 15m/s.

Field	**Area (hectares)**	**Average Ground Sampling Distance (cm)**	**Keypoint match density (#/image)**	**Average Density Point cloud (per m^3)**	**Mean Reprojection Error (pixels)**
Grass Tr1	2.2	1.23	34558.4	n/a	0.24
Grass Tr2	4.0	1.25	36931.7	n/a	0.25
Grass Tr3	3.2	1.23	32025.4	n/a	0.26
Grass Tr4	1.6	1.27	37356.9	n/a	0.24
Wheat Tr1	4.1	2.12	27923.7	901.41	0.17
Wheat Tr2	2.4	2.08	30704.8	788.34	0.17
Wheat Tr3	2.6	2.23	25539.8	1415.57	0.16
Wheat Tr4	1.8	2.18	20610.1	681.98	0.17

Table S7. Quality report numbers for each of the selected grass and wheat test fields separately that were retrieved by processing the raw UAV images in PIX4Dmapper version 4.8.4 with a pixel size of 5cm. The UAV images were collected in May 2023 for grassland and August 2023 for wheatland in Boo, Hjortkvarn Municipality, Sweden, using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters, and a speed of 15m/s.

Field	**Area (hectares)**	**Average Ground Sampling Distance (cm)**	**Keypoint match density (#/image)**	**Average Density Point cloud (per m^3)**	**Mean Reprojection Error (pixels)**
Grass Te1	3.1	1.24	34784.5	n/a	0.24
Grass Te2	1.0	1.22	30895.6	n/a	0.24
Grass Te3	1.3	1.38	30252.9	n/a	0.26
Grass Te4	3.1	1.16	28126.1	n/a	0.26
Wheat Te1	1.9	2.27	28775.6	817.68	0.17
Wheat Te2	2.4	2.15	28840.9	1355.76	0.16
Wheat Te3	2.9	2.26	28620.9	1076.55	0.16
Wheat Te4	2.4	2.31	27977.5	1288.14	0.16

9.2 Object-based and Pixel-based Classifications

Table S8. Jeffries-Matusita scores of the RGB mean derived texture band for both grass and wheatland, obtained by applying the JMdist method from R-package Varsel (Dalponte & Oerka, 2021). The RGB mean band obtained the highest total separability score out of all the 10 classification bands.

Combination	Wheatland	Grassland
Dry Damage vs New Damage/Grass	1.41	1.32
Dry Damage vs Shadow/ Dark Damage	1.41	1.41
Dry Damage vs Grass/Wheat	1.22	1.30
New Damage/ Grass vs Shadow/ Dark Damage	0.04	1.05
New Damage/Grass vs Grass/Wheat	1.08	0.22
Shadow/Grass vs Grass/Wheat	1.08	1.20

Table S9. Top 5 bands with the highest average Jeffries-Matusita scores, obtained by applying the JMdist method from the R-package Varsel (Dalponte & Oerka, 2021) for both the wheat and grasslands data.

Grass Bands	Mean Score	Wheat Bands	Mean Score
RGB mean	1.08	RGB mean	1.04
Red	0.99	Red	0.83
Green	0.98	Blue	0.81
Blue	0.82	Green	0.77
Re	0.81	NDVI	0.75

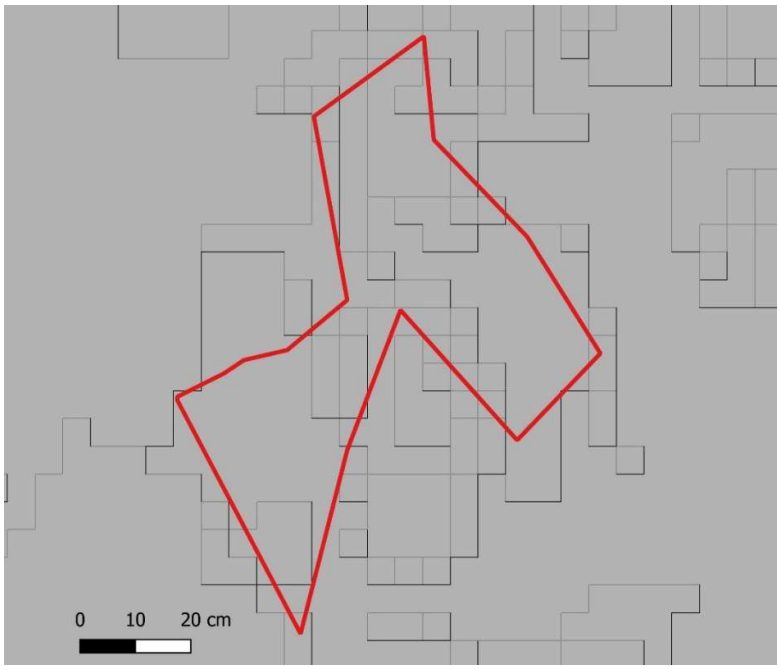

Figure S1. This figure represents a snapshot of a segmented image created using a spectral range of 0.05, spatial range of 10, and minimum size set to 0 at the segmentation function provided by the R-package SegOptim (Gonçalves et. al., 2019). The test data polygon shown in the red triangle represents one damage spot and shows that within the one damage spot, several one/few pixel segmented objects exist.

Table S10. Top 5 hyperparameters for the grassland and wheatland classifications. The spectral range, spatial range, and minimum size are hyperparameters that had to be set in the segmentation method provided by R-package SegOptim (Gonçalves et. al., 2019).

Hyperparameter	**Grassland RF**	**Grassland SVM**	**Wheatland RF**	**Wheatland SVM**
Spectral Range				
1.	0.01	0.04	0.01	0.02
2.	0.02	0.04	0.02	0.02
3.	0.02	0.03	0.02	0.01
4.	0.01	0.03	0.01	0.03
5.	0.02	0.02	0.02	0.03
Spatial Range				
1.	10	10	10	10
2.	10	10	10	10
3.	10	10	10	10
4.	10	10	10	10
5.	10	10	10	10
Minimum Size				
1.	20	25	20	20
2.	25	20	25	15
3.	20	20	20	15
4.	25	25	25	25
5.	15	20	15	20
Kappa				
1.	0.97	0.96	0.97	0.97
2.	0.96	0.96	0.96	0.97
3.	0.96	0.96	0.96	0.97
4.	0.96	0.96	0.96	0.97
5.	0.96	0.95	0.96	0.97

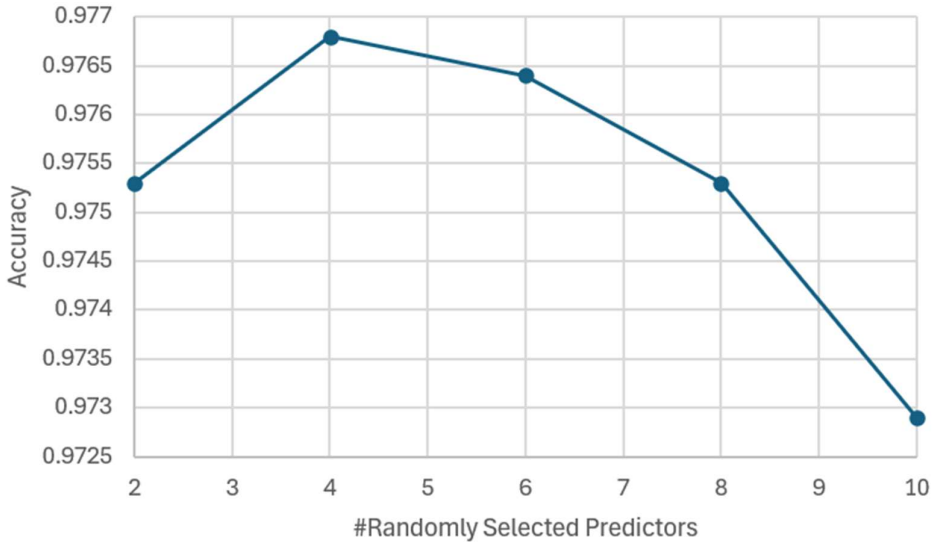

Figure S2. Grassland's Random Forest pixel-based hyperparameter tuning performances visualized for the different number of Randomly Selected Predictors. The x-axis shows the number of Randomly Selected Predictors and the y-axis the performance in the form of an Accuracy based on a 10-fold cross-validation.

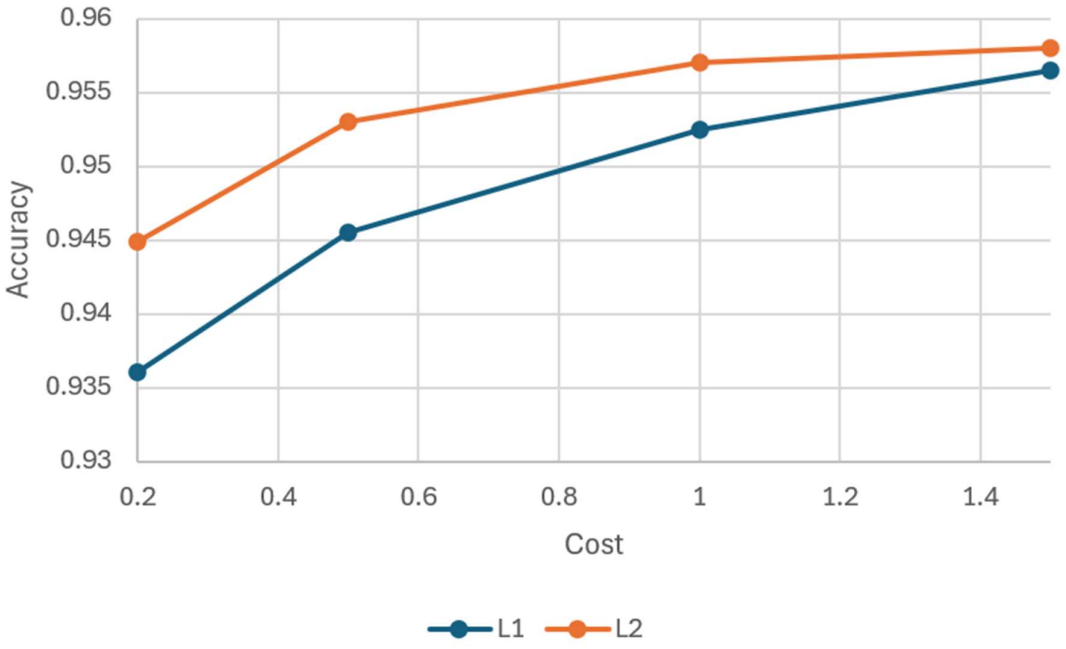

Figure S3. Grassland Support Vector Machine pixel-based hyperparameter tuning performances for the different costs and Losses. The x-axis represents the cost values and the y-axis the performance in the form of an Accuracy based on a 10-fold cross-validation. The red line represents the L1 loss function and the blue line the L2 loss function.

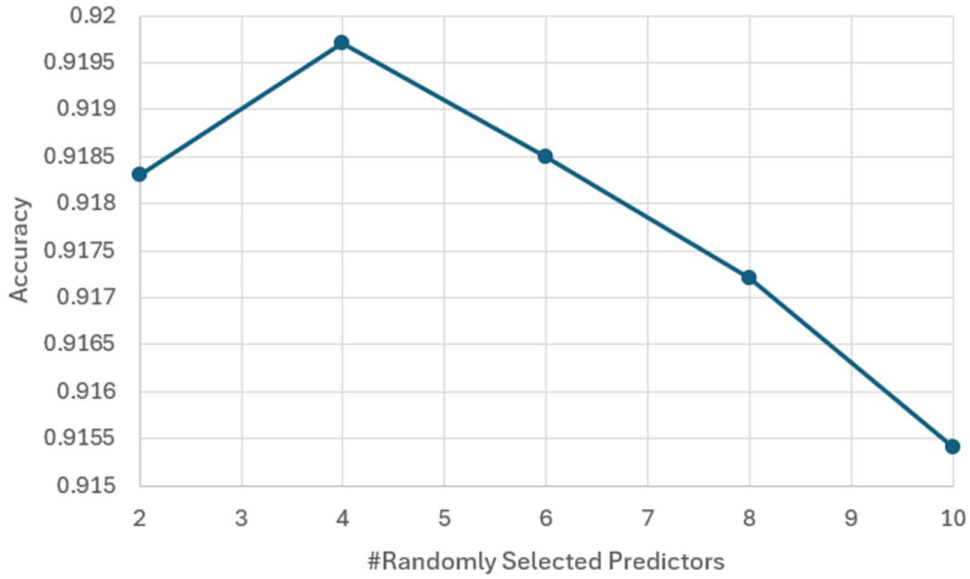

Figure S4. Wheatland Random Forest pixel-based hyperparameter tuning performances for the different number of Randomly Selected Predictors. The x-axis shows the number of Randomly Selected Predictors and the y-axis the performance in the form of an Accuracy based on a 10-fold cross-validation.

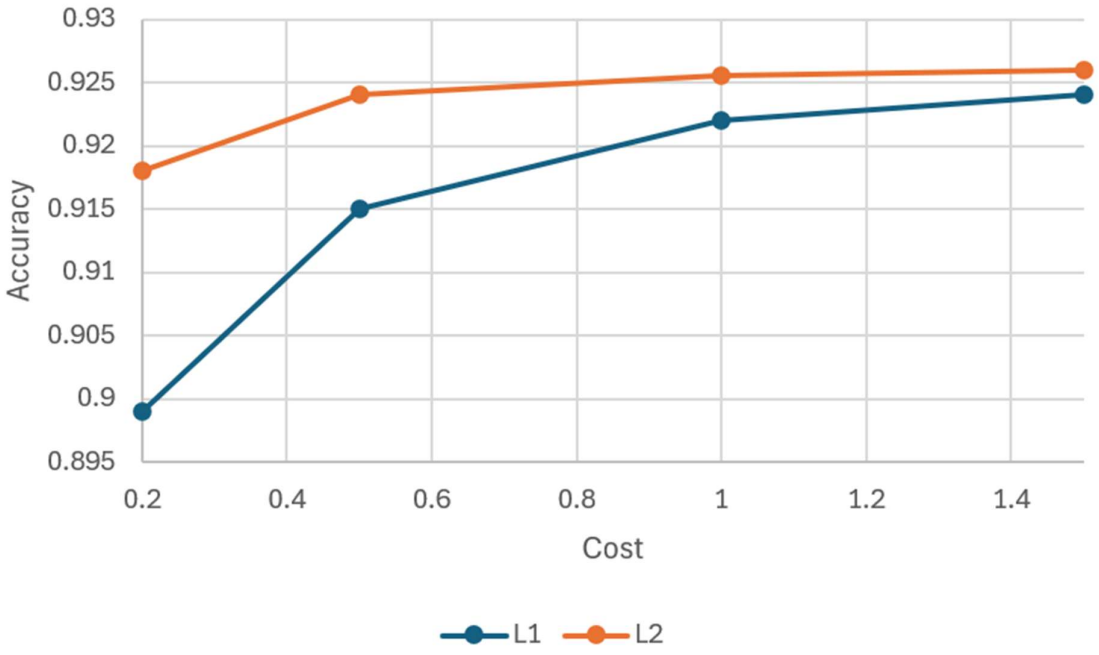

Figure S5. Wheatland Support Vector Machine pixel-based hyperparameter tuning performances for the different costs and Losses. The x-axis represents the cost values and the y-axis the performance in the form of an Accuracy based on a 10-fold cross-validation. The red line represents the L1 loss function and the blue line the L2 loss function.

Table S11. The average scores for the kappa and overall accuracy of the 10-fold cross-validation of the training of the object-based and pixel-based classifiers for the classification of the grassland train fields.

	Grass Object RF	**Grass Object SVM**	**Grass Pixel RF**	**Grass pixel SVM**
Kappa	0.89	0.86	0.97	0.94
Overall Accuracy	0.93	0.90	0.98	0.96

Table S12. The average scores for the kappa and overall accuracy of the 10-fold cross-validation of the training of the object-based and pixel-based classifiers for the classification of the wheatland train fields.

	Wheat Object RF	**Wheat Object SVM**	**Wheat Pixel RF**	**Wheat Pixel SVM**
Kappa	0.84	0.87	0.88	0.89
Overall Accuracy	0.89	0.91	0.92	0.93

Table S13. The number of agreements between the pixel and object-based classifications obtained by the trained pixel and object-based classifications of the grassland fields. Field Test 4 is the field closest to the median.

#Agreements RF	**Field Number RF**	**#Agreements SVM**	**Field Number SVM**
2398473	Test 3	2217896	Test 3
3943603	Test 2	3711459	Test 2
4616802	Train 4	4259366	Train 4
6907405	Train 1	6667948	Train 1
8512255	Test 4	8605484	Test 4
9708142	Train 3	9903089	Train 3
10648851	Test 1	10140052	Test 1
11815157	Train 2	10927414	Train 2

Table S14. The number of agreements between the pixel and object-based classifications obtained by the trained pixel and object-based classifications of the wheatland fields. Field Train 3 is the field closest to the median.

#Agreements RF	Field Number RF	#Agreements SVM	Field Number SVM
4385040	Test 4	4278099	Test 4
5747440	Test 3	6186529	Train 1
5800534	Test 1	6347175	Train 3
6018725	Train 1	6457279	Test 1
6259974	Train 3	6553377	Test 3
7842449	Train 2	8036223	Train 2
7957603	Test 2	8167041	Test 2
11984389	Train 4	12546766	Train 4

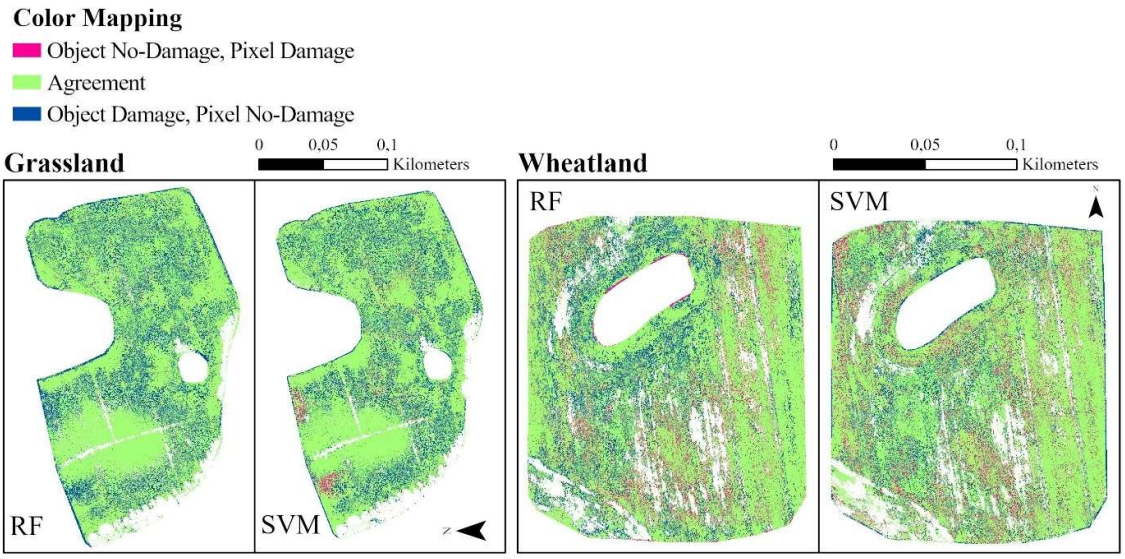

Figure S6, Difference maps of the grass field test 4 (left) and wheat field train 3 (right) of the object and pixel-based classifications in coordinate reference system WGS 84 / UTM zone 33N. The white areas in the fields, representing shadow/grass, were removed from the analysis. The damage class represents both Dry and Dew/Dark damage, the No-Damage represents Grass/Wheat. The data from the grassland was collected in May 2023 and the wheatland in August 2023 in Boo, Hjortkvarn Municipality, Sweden using a UAV DJI Mavic 3 Multispectral (DJI, n.d.) that had a flight height of 40 meters and a speed of 15m/s.

9.3 Point Cloud Height Threshold Classification

Table S15. Top ten optimal hyperparameters according to the hyperparameter tuning done for the point cloud height threshold classification method.

Classification Threshold	Power of proportional distance	Number of neighbors	Dice
1.00E-05	1	80	0.28
1.00E-05	1	50	0.25
2.50E-05	1	80	0.25
2.50E-05	1	50	0.22
5.00E-05	1	80	0.22
1.00E-05	1	20	0.21
7.50E-05	1	80	0.20
1.00E-05	2	80	0.20
5.00E-05	1	50	0.19
1.00E-05	2	50	0.19

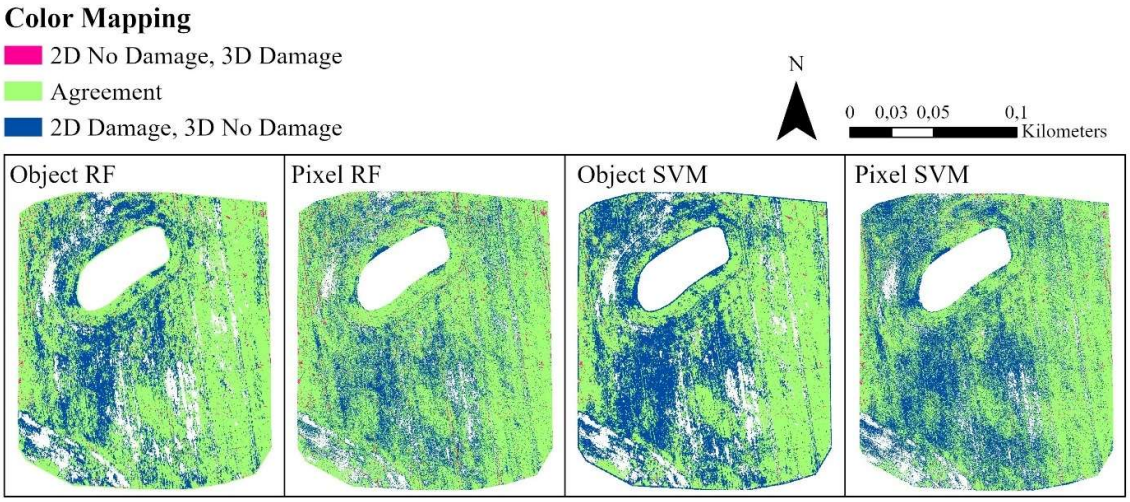

Figure S7. Difference maps of the wheatland 2D classifications and the Point Cloud Height Threshold classification for the wheat field train 3 in coordinate reference system WGS 84 / UTM zone 33N. The grass is removed from the analysis and shown in white. The data was collected in August 2023 in Boo, Hjortkvarn Municipality, Sweden using a UAV DJI Mavic 3 Multispectral (DJI, n.d.), flight height of 40 meters and a speed of 15m/s.

9.4. R-packages

Table S16. This table represents an overview of all the R-packages used to develop the classification methods (i.e. object-based, pixel-based, and point cloud height threshold classification) for the damage mappings in agricultural grass and wheatlands.

Name	Reference	Where Used
caret	Kuhn, 2008	- Pixel-based Classification
data.table	v1.14.8; Dowle & Srinivasan, 2023	- Pixel-based Classification - Object-based Classification - Point Cloud Height Threshold Classification
devtools	v2.4.5; Wickham et al. 2022	- Pixel-based Classification
doParallel	v1.0.17; Weston & Microsoft Corporation, 2022	- Pixel-based Classification - Object-based Classification
dplyr	v1.1.3; Wickham et al., 2023	- Pixel-based Classification - Object-based Classification
elevatr	Hollister et al., 2023	- Preprocessing: Object-based and Pixel-based Classification
foreach	v1.5.2; Weston & Microsoft, 2022	- Object-based Classification
furrr	v0.3.1; Vaughan & Dancho, 2022	- Object-based Classification - Point Cloud Height Threshold Classification
ggplot2	Wickham, 2016	- Preprocessing: Object-based and Pixel-based Classification - Object-based Classification - Point Cloud Height Threshold Classification
GLCMTextures	Ilich, 2020	- Preprocessing: Object-based and Pixel-based Classification
grDevices	R Core Team, 2022	- Pixel-based Classification
gtools	v3.9.5; Warnes et al., 2023	- Pixel-based Classification
htmlwidgets	v1.6.4; Vaidyanathan et al., 2023	- Pixel-based Classification
leafsync	v01.0; Appelhans & Russell, 2019	- Pixel-based Classification
LiblineaR	v2.10-23; Helleputte et al. 2023	- Pixel-based Classification
lidR	Roussel et al., 2020; v4.1.1; Roussel & Auty, 2024	- Point Cloud Height Threshold Classification
mapview	v2.11.2; Appelhans et al.,2023	- Pixel-based Classification - Object-based Classification
MLmetrics	v1.1.1; Yan, 2016	- Pixel-based Classification - Object-based Classification
parallel	R Core Team, 2022	- Pixel-based Classification
plotly	Sievert, 2020	- Pixel-based Classification
pryr	v0.1.6; Wickham, 2023	- Object-based Classification
purrr	v1.0.2; Wickham & Henry, 2023	- Object-based Classification

		- Point Cloud Height Threshold Classification
randomForest	Liaw & Wiener, 2002	- Pixel-based Classification - Object-based Classification
ranger	Wright & Ziegler, 2017	- Pixel-based Classification
raster	v3.6-23; Hijmans, 2023	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification - Object-based Classification - Point Cloud Height Threshold Classification
rasterVis	v0.51.6; Lamigueiro & Hijmans, 2023	- Pixel-based Classification - Object-based Classification
RColorBrewer	v1.1-3; Neuwirth, 2022	- Pixel-based Classification
Rcpp	v1.0.11; Eddelbuettel et al., 2023	- Pixel-based Classification
RCSF	v1.0.2; Roussel & Qi, 2020	- Point Cloud Height Threshold Classification
rgdal	v1.6-7; Bivand & Rowlingson, 2023	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification - Object-based Classification - Point Cloud Height Threshold Classification
RStoolbox	v0.4.0; Leutner et al., 2024	- Pixel-based Classification - Object-based Classification
SegOptim	v0.2.5; Goncalves, 2022	- Object-based Classification
sf	Pebesma, 2018; Pebesma & Bivand, 2023	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification - Object-based Classification - Point Cloud Height Threshold Classification
snow	v0.4-4: Tierney et al., 2021	- Pixel-based Classification
Sp	Pebesma & Bivand, 2005; Bivand et al., 2013	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification - Object-based Classification
stringr	v1.5.1; Wickham, 2023	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification - Object-based Classification
terra	v1.7-39; Hijmans, 2023	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification - Object-based Classification - Point Cloud Height Threshold Classification

tidyterra	Hernangómez, 2023	- Preprocessing: Object-based and Pixel-based Classification
Tidyverse	Wickham et al., 2019	- Preprocessing: Object-based and Pixel-based Classification - Pixel-based Classification
webshot	v0.5.5; Chang, 2023	- Pixel-based Classification
whitebox	Lindsay, 2016; v2.2.0; Wu & Brown, 2022	- Preprocessing: Object-based and Pixel-based Classification

9.5 References

Appelhans, T., Detsch, F., Reudenbach, C., Woellauer, S. (2023*). _mapview: Interactive Viewing of Spatial [R package version 2.11.2]*. https://CRAN.R-project.org/package=mapview

Appelhans T, Russell K (2019*). _leafsync: Small Multiples for Leaflet Web Maps_ [R package version 0.1.0]*. https://CRAN.R-project.org/package=leafsync

Bivand, R. S., Pebesma, E., & Gomez-Rubio, V. (2013). *Applied spatial data analysis with R, Second edition*. https://asdar-book.org/

Bivand. R., Keitt, T., & Rowlingson, B. (2023). *rgdal: Bindings for the 'Geospatial' Data Abstraction Library [R package version 1.6-7]*. https://CRAN.R-project.org/package=rgdal

Chang, W. (2023). *_webshot: Take Screenshots of Web Pages_ [R package version 0.5.5]*. https://CRAN.R-project.org/package=webshot

Weston, S., Microsoft Corporation (2022). *_doParallel: Foreach Parallel Adaptor for the 'parallel' Package_ [R package version 1.0.17]*. https://CRAN.R-project.org/package=doParallel

Dowle, M., Srinivasan, A. (2023). *_data.table: Extension of `data.frame`_ [R package version 1.14.8]*. https://CRAN.R-project.org/package=data.table

Eddelbuettel D, Francois R, Allaire J, Ushey K, Kou Q, Russell N, Ucar I, Bates D, Chambers J (2023). *_Rcpp: Seamless R and C++ Integration_ [R package version 1.0.11]*. https://CRAN.R-project.org/package=Rcpp

Goncalves, J. (2022). *_SegOptim: Optimization of Image Segmentation Parameters for Object-Based Image Analysis (OBIA)_ [R package version 0.2.5]*. https://github.com/joaofgoncalves/SegOptim

Helleputte, T., Gramme, P., Paul, J.(2023). *LiblineaR: Linear Predictive Models Based On The Liblinear C/C++ Library [R package version 2.10-23]*. *https://www.csie.ntu.edu.tw/~cjlin/liblinear/*

Hernangómez, D. (2023). Using the tidyverse with terra objects: the tidyterra package. *Journal of Open Source Software*, *8*(91), 5751. https://doi.org/10.21105/joss.05751

Hijmans, R. J. (2023). *raster: Geographic Data Analysis and Modeling [R package version 3.6-23]*. Retrieved from https://CRAN.R-project.org/package=raster

Hijmans, R. J. (2023). *terra: Spatial Data Analysis [R package version 1.7-39]*. https://CRAN.R-project.org/package=terra

Hollister, J., Shah, T., Nowosad, J., Robitaille, A. L., Beck, M. W., & Johnson, M. (2023). *elevatr: Access Elevation Data from Various APIs*. https://doi.org/10.5281/zenodo.8335450

Ilich, A. R. (2020). *GitHub - ailich/GLCMTextures: This R package calculates the most common gray-level co-occurrence matrix (GLCM) texture metrics used for spatial analysis on raster data.* https://github.com/ailich/GLCMTextures

Kuhn, M. (2008). Building Predictive Models inRUsing theCaretPackage. *Journal of Statistical Software*, *28*(5). https://doi.org/10.18637/jss.v028.i05

Lamigueiro, O. P., & Hijmans, R. (2023) *RasterVis [R package version 0.51.6]*. https://oscarperpinan.github.io/rastervis/

Leutner, B., Horning N, Schwalb-Willmann J (2024*). _RStoolbox: Remote Sensing Data Analysis [R package version 0.4.0]*. https://CRAN.R-project.org/package=RStoolbox

Liaw, A., & Wiener, M. (2002). CRAN: R News. *R News*, *2*(3), 18–22. https://cran.r-project.org/doc/Rnews/

Lindsay, J. (2016b). Whitebox GAT: A case study in geomorphometric analysis. *Computers & Geosciences*, *95*, 75–84. https://doi.org/10.1016/j.cageo.2016.07.003

Neuwirth, E., (2022). *_RColorBrewer: ColorBrewer Palettes_[R package version 1.1-3]*. https://CRAN.R-project.org/package=RColorBrewer

Pebesma, E. J., & Bivand, R. (2005, November). Classes and methods for spatial data in R. *R News*, Vol. 5, pp. 9–13. https://CRAN.R-project.org/doc/Rnews/

Pebesma, E. (2018). Simple features for R: Standardized support for spatial vector data. *The R Journal*, *10*(1), 439-446. https://doi.org/10.32614/rj-2018-009

Pebesma, E., & Bivand, R. (2023). *Spatial Data Science* (1st ed.). Chapman & Hall. https://doi.org/10.1201/9780429459016

R Core Team (2022). *R: A language and environment for statistical computing.* R Foundation for Statistical Computing. https://www.R-project.org/

Roussel, J. R., Auty, D., Coops, N. C., Tompalski, P., Goodbody, T. R., Meador, A. S., Bourdon, J. F., De Boissieu, F., & Achim, A. (2020). lidR: An R package for analysis

of Airborne Laser Scanning (ALS) data. *Remote Sensing of Environment*, *251*, 112061. https://doi.org/10.1016/j.rse.2020.112061

Roussel, J. R. & Auty, D. (2024). *Airborne LiDAR Data Manipulation and Visualization for Forestry Applications [R package version 4.1.1.]*. https://cran.r-project.org/package=lidR

Roussel, J. & Qi, J. (2020). *_RCSF: Airborne LiDAR Filtering Method Based on Cloth Simulation_ [R package version 1.0.2]*. https://CRAN.R-project.org/package=RCSF

Sievert, C. (2020). *Interactive Web-based Data Visualization with R, Plotly, and Shiny*. Chapman & Hall/CRC The R Series. https://plotly-r.com/

Tierney L, Rossini AJ, Li N, Sevcikova H (2021). *_snow: Simple Network of Workstations_ [R package version 0.4-4]*. https://CRAN.R-project.org/package=snow

Vaidyanathan R, Xie Y, Allaire J, Cheng J, Sievert C, Russell K (2023). *_htmlwidgets: HTML Widgets for R_ [R package version 1.6.4]*. https://CRAN.R-project.org/package=htmlwidgets

Vaughan D, Dancho M (2022). *_furrr: Apply Mapping Functions in Parallel using Futures_ [R package version 0.3.1]*. https://CRAN.R-project.org/package=furrr

Warnes G, Bolker B, Lumley T, Magnusson A, Venables B, Ryodan G, Moeller S (2023). *_gtools: Various R Programming Tools_ [R package version 3.9.5]*. https://CRAN.R-project.org/package=gtools

Weston, S., Microsoft (2022). *_foreach: Provides Foreach Looping Construct_ [R package version 1.5.2]*. https://CRAN.R-project.org/package=foreach

Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer. https://ggplot2.tidyverse.org

Wickham H (2023). *_pryr: Tools for Computing on the Language_ [R package version 0.1.6]*. https://CRAN.R-project.org/package=pryr

Wickham H (2023). *_stringr: Simple, Consistent Wrappers for Common String Operations_ [R package version 1.5.1]*. https://CRAN.R-project.org/package=stringr

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T., Miller, E., Bache, S., Müller, K., Ooms, J., Robinson, D., Seidel, D., Spinu, V., . . . Yutani, H. (2019). Welcome to the Tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wickham H, François R, Henry L, Müller K, Vaughan D (2023). *_dplyr: A Grammar of Data Manipulation_ [R package 1.1.3]*. https://CRAN.R-project.org/package=dplyr

Wickham, H. & Henry, L. (2023). *_purrr: Functional Programming Tools_ [R package version 1.0.2]*. https://CRAN.R-project.org/package=purrr

Wickham, H., Hester, J., Chang, W., & Bryan, J. (2022). *devtools: Tools to Make Developing R Packages Easier [R package version 2.4.5]*. https://CRAN.R-project.org/package=devtools

Wright, M. N., & Ziegler, A. (2017). ranger: A Fast Implementation of Random Forests for High Dimensional Data in C++ and R. *Journal of Statistical Software*, *77*(1). https://doi.org/10.18637/jss.v077.i01

Wu, Q., & Brown, A. (2022). *'WhiteboxTools': 'WhiteboxTools' R Frontend [R package version 2.2.0*.] https://CRAN.R-project.org/package=whitebox

Yan, Y. (2016). *_MLmetrics: Machine Learning Evaluation Metrics_ [R package version 1.1.1]*. https://CRAN.R-project.org/package=MLmetrics