

Eduardo Garcia do Nascimento

Productive Crop Field Detection

Sorocaba, SP

January 15th 2024

Eduardo Garcia do Nascimento

Productive Crop Field Detection

Master's dissertation presented for the Graduate Program in Computer Science (PPGCC-So) at the Federal University of São Carlos as part of the requirements for obtaining the Master's degree in Computer Science. Research focus: Remote Sensing and Artificial Intelligence.

Universidade Federal de São Carlos – UFSCar

Centro de Ciências em Gestão e Tecnologia – CCGT

Programa de Pós-Graduação em Ciência da Computação – PPGCC-So

Supervisor: Prof. Dr. Tiago A. Almeida

Co-supervisor: Prof. Dr. Jurandy G. Almeida Jr.

Sorocaba, SP

January 15th 2024

Garcia do Nascimento, Eduardo

Productive Crop Field Detection/ Eduardo Garcia do Nascimento. – 2024.

186 f. : 30 cm.

Dissertação (Mestrado) – Universidade Federal de São Carlos – UFSCar

Centro de Ciências em Gestão e Tecnologia – CCGT

Programa de Pós-Graduação em Ciência da Computação – PPGCC-So.

Supervisor: Prof. Dr. Tiago A. Almeida

Banca examinadora: Prof. Dr. Tiago A. Almeida, Prof. Dr. Jefersson Alex dos Santos, Prof. Dr. João Paulo Papa

Bibliografia

1. Detecção de Talhões Produtivos. 2. Aprendizado Contrastivo. 3. Redes Neurais Siamesas. I. Eduardo Garcia do Nascimento. II. Universidade Federal de São Carlos. III. Título



UNIVERSIDADE FEDERAL DE SÃO CARLOS

Centro de Ciências em Gestão e Tecnologia
Programa de Pós-Graduação em Ciência da Computação

Relatório de Defesa de Dissertação

Candidato: Eduardo Garcia do Nascimento

Aos 15/01/2024, às 10:00, realizou-se na Universidade Federal de São Carlos, nas formas e termos do Regimento Interno do Programa de Pós-Graduação em Ciência da Computação, a defesa de dissertação de mestrado sob o título: Productive Crop Field Detection, apresentada pelo candidato Eduardo Garcia do Nascimento. Ao final dos trabalhos, a banca examinadora reuniu-se em sessão reservada para o julgamento, tendo os membros chegado ao seguinte resultado:

Participantes da Banca

Prof. Dr. Tiago Agostinho de Almeida
Prof. Dr. Jefersson Alex dos Santos
Prof. Dr. João Paulo Papa

Função	Instituição	Resultado	Resultado Final
Presidente	UFSCar	_____	_____
Titular	University Of	_____	_____
Titular	Sheffield UFSCar	_____	_____

Parecer da Comissão Julgadora*:

Encerrada a sessão reservada, o presidente informou ao público presente o resultado. Nada mais havendo a tratar, a sessão foi encerrada e, para constar, eu, Tiago Agostinho de Almeida, representante do Programa de Pós-Graduação em Ciência da Computação, lavrei o presente relatório, assinado por mim e pelos membros da banca examinadora.

Prof. Dr. Tiago Agostinho de Almeida

Representante do PPG: Tiago Agostinho de Almeida

Prof. Dr. Jefersson Alex dos Santos

Prof. Dr. João Paulo Papa

Certifico que a defesa realizou-se com a participação à distância do(s) membro(s) Tiago Agostinho de Almeida, Jefersson Alex dos Santos, João Paulo Papa e, depois das arguições e deliberações realizadas, o(s) participante(s) à distância está(ão) de acordo com o conteúdo do parecer da banca examinadora redigido neste relatório de defesa.

Prof. Dr. Tiago Agostinho de Almeida

Não houve alteração no título Houve alteração no título. O novo título passa a ser:

Observações:

a) Se o candidato for reprovado por algum dos membros, o preenchimento do parecer é obrigatório.

b) Para gozar dos direitos do título de Mestre ou Doutor em Ciência da Computação, o candidato ainda precisa ter sua dissertação ou tese homologada pelo Conselho de Pós-Graduação da UFSCar.



DECLARAÇÃO

Eu, **Jefersson Alex dos Santos**, declaro para os devidos fins que irei participar da banca de defesa de dissertação do aluno **Eduardo Garcia do Nascimento**, através do meio de comunicação eletrônica síncrona, sendo a Ata do referido ato assinada em meu nome pelo presidente da sessão.

Jefersson A. dos Santos

Jefersson Alex dos Santos
Sheffield, UK, 16 de Janeiro de 2023

To my beloved parents, José de Aquino and Vera Lúcia Garcia

Acknowledgements

I would like to express my heartfelt gratitude to the following individuals and entities, without whom the completion of this dissertation would not have been possible:

First and foremost, I am profoundly grateful to God for granting me the strength, health, wisdom, and perseverance throughout this academic journey. Your divine guidance and blessings have been my source of inspiration and motivation, and I am thankful for the opportunity you have bestowed upon me.

I extend my deepest appreciation to Prof. Dr. Tiago Almeida, my advisor, for his unwavering support, mentorship, and encouragement. His guidance and belief in my potential played a pivotal role in my decision to pursue the master's degree program. I am also indebted to Prof. Dr. Jurandy Almeida for joining my academic journey as my co-advisor, offering valuable insights, and contributing significantly to the development of this dissertation.

I am indebted to John Just, the Data Science and Analytics expert from John Deere, for his invaluable assistance and support. His expertise, provided regardless of the time or day, even during late hours on weekends, has been instrumental in shaping my research and enriching my understanding of the subject matter.

My sincere thanks go to Marlon Adamy, my previous manager at John Deere during the most time of this journey, for his unwavering support, encouragement, and belief in my abilities. I am also grateful to Peter Muench, my current manager, for motivating me and demonstrating unconditional support to finish this study while transitioning to my new role.

To my parents, I will express my gratitude in Portuguese so that they fully understand: Sou profundamente grato aos meus pais, Vera Lúcia Garcia e José de Aquino, pela paciência e suporte incondicional. Sempre ao meu lado, me apoiando mesmo em situações de ausência devido aos meus compromissos acadêmicos. O amor e compreensão deles foi essencial pra tornar essa jornada possível.

To everyone mentioned above and those who supported me in various ways, I offer my heartfelt thanks.

*“The future belongs to those who believe in the beauty of their dreams.”
(Eleanor Roosevelt)*

Resumo

Em agricultura de precisão, detectar e delinear talhões produtivos é uma prática essencial que permite ao agricultor avaliar o desempenho operacional separadamente e comparar diversas variedades de sementes, defensivos e fertilizantes. Entretanto, a identificação manual de talhões produtivos é frequentemente demorada, cara e subjetiva. Estudos anteriores exploraram diferentes métodos de detecção de talhões utilizando algoritmos avançados de aprendizado de máquina para dar suporte às decisões de especialistas, entretanto, eles frequentemente enfrentam limitações devido à baixa disponibilidade de dados rotulados de alta qualidade. Neste contexto, é proposto um sistema de detecção de talhões produtivos baseados em um conjunto de dados de alta qualidade gerado a partir da combinação de dados operacionais de máquinas agrícolas com imagens de satélite Sentinel-2 extraídos ao longo do tempo. No melhor do nosso conhecimento, este é o primeiro trabalho científico que supera os desafios impostos pela segmentação de campos produtivos utilizando esta combinação de técnicas. Em seguida, são apresentados três métodos, utilizando como base o estado da arte em métodos supervisionados e auto-supervisionados, selecionados de acordo com as características da base, para a detecção automática de talhões produtivos. Adicionalmente, são reportados resultados com alta acurácia utilizando o aprendizado com exemplos positivos e não rotulados, o qual se encaixa perfeitamente neste cenário, onde temos alta confiança nos dados positivos. Por fim, melhores desempenhos foram obtidos com o Aprendizado Contrastivo considerando sua característica de aumento de dados que permite o treinamento com uma quantidade maior de amostras mesmo que artificialmente geradas.

Palavras-chaves: Agricultura de Precisão, detecção de talhões, aprendizado contrastivo.

Abstract

In precision agriculture, detecting productive crop fields is an essential practice that allows the farmer to evaluate operating performance separately and compare different seed varieties, pesticides, and fertilizers. However, manually identifying productive fields is often time-consuming, costly, and subjective. Previous studies explore different methods to detect crop fields using advanced machine learning algorithms to support the specialists' decisions, but they often lack good quality labeled data. In this context, we propose a framework for productive crop field detection based on high-quality dataset generated by machine operation combined with Sentinel-2 images tracked over time. As far as we know, it is the first one to overcome the lack of labeled samples by using this combination of techniques. In sequence, we present three methods, based on state-of-the-art supervised and self-supervised methods, selected according to the dataset characteristics, to detect productive crop fields. Finally, we demonstrate high accuracy results in Positive Unlabeled learning, which perfectly fits the problem where we have high confidence in the positive samples. Finally, best performances have been found with Contrastive Learning, given its ability to augment data, allowing the model to be trained with a larger dataset considering the artificially created samples.

Keywords: Precision Agriculture, crop field detection, contrastive learning.

List of Figures

Figure 1 – Two crop fields highlighted in a satellite image. Source: own authorship.	25
Figure 2 – End-to-end diagram of the proposed framework. Source: own authorship.	26
Figure 3 – Geohash grid system and leveling. Source: Geospatial World website ¹ .	30
Figure 4 – Quadkey codes according to location and detail level. Source: Microsoft Bing website.	31
Figure 5 – Overlapping of radio-based systems using a triangular grid system. Source: Kumari e Singh (2012).	32
Figure 6 – The square, hexagonal and triangular grids with their symmetric coordinate systems. Source: Nagy (2022).	33
Figure 7 – The segmentation process of the watershed algorithm. Source: Zheng et al. (2021).	34
Figure 8 – Simplified example of a GrabCut segmentation process. Source: Xiao et al. (2017).	34
Figure 9 – Neural network basic forward propagation. Source: LeCun, Bengio e Hinton (2015).	37
Figure 10 – Neural network backpropagation. Source: LeCun, Bengio e Hinton (2015).	38
Figure 11 – Parallel bagging used in PU learning. Source: Denis, Gilleron e Letouzey (2005).	39
Figure 12 – Steps of a 2D Convolutional Neural Networks. Source: Li et al. (2022b).	41
Figure 13 – Triplet loss learning process. Source: Schroff, Kalenichenko e Philbin (2015).	41
Figure 14 – Data augmentation examples applied to an image. Source: Le-Khac, Healy e Smeaton (2020).	43
Figure 15 – Segmentation results on sample images. Source: Minaee et al. (2021). .	44
Figure 16 – Semantic segmentation sample with the original image on the left and segmented output on the right. Source: Carvalho et al. (2021).	45
Figure 17 – Instance segmentation sample with original image on the left and segmented output on the right. Source: Carvalho et al. (2021).	46
Figure 18 – Panoptic segmentation example. The original image is on the left, and the segmented output is on the right. Source: Carvalho et al. (2021). .	47
Figure 19 – Comparison between the ideal segmentation and one generated by the CGRG method. Source: Evans et al. (2002).	51
Figure 20 – Segmentation by grouping super pixels and machine learning classification. Source: Garcia-Pedrero, Gonzalo-Martin e Lillo-Saavedra (2017).	52

Figure 21 – Visual comparison between the segmentation results obtained with CNN and gPb-UCM. Source: Garcia-Pedrero, Gonzalo-Martin e Lillo-Saavedra (2017).	53
Figure 22 – Segmentation results obtained by Fractal-ResUNet method. Source: Waldner et al. (2021).	55
Figure 23 – Multi-Swin Mask Transformer comparison results. Source: Zhong et al. (2023).	56
Figure 24 – NDVI patterns comparison between a productive and a non-productive area. Source: own authorship.	62
Figure 25 – Sequence to build a sample based upon the hexagon time series. Source: own authorship.	62
Figure 26 – Automatic labeling of hexagons based on agricultural machines operation. Source: own authorship.	63
Figure 27 – Map view of the crop fields employed in this study. Source: own authorship.	64
Figure 28 – Distribution of the number of samples by satellite image date. Source: own authorship.	64
Figure 29 – Productive crop field filled with positively labeled hexagons in green surrounded by a three hexagons layer of inferred negative samples. Source: own authorship.	65
Figure 30 – A non-productive field properly classified with red hexagons. Source: own authorship.	66
Figure 31 – Multitemporal time series sample visualization with 16 rows (i.e., each row corresponding to a unique date a satellite image was captured of the hexagon location) and 12 columns (i.e., satellite band values). Source: own authorship.	67
Figure 32 – Methods to build samples based on months’ distribution. Source: own authorship.	67
Figure 33 – PU Learning architecture employed in the benchmark testing. Source: own authorship.	70
Figure 34 – Triplet Loss Siamese Networks architectures employed in the benchmark testing. Source: own authorship.	71
Figure 35 – Contrastive Learning architectures employed in the benchmark testing. Source: own authorship.	72
Figure 36 – Leave One Out iteration report. Source: own authorship.	73
Figure 37 – Method consistency when classifying crop fields. Source: own authorship.	74
Figure 38 – Blue hexagons represent prediction mistakes, usually found between productive and non-field areas.	78

List of Tables

Table 1 – Related work summary containing most important points of each study.	58
Table 2 – Productive Fields Dataset Description.	65
Table 3 – Results achieved by the PU Learning model.	74
Table 4 – Results achieved by Triplet Loss Siamese model.	75
Table 5 – Results achieved by SimCLR model.	76
Table 6 – Results achieved by the outcome prediction models.	77

List of abbreviations and acronyms

ABNT	Brazilian Association of Technical Norms
AI	Artificial Intelligence
ANN	Artificial Neural Network
BDE	Boundary Displacement Error Index
CGRG	Canonically Guided Region Growing
CNN	Convolutional Neural Network
DL	Deep Learning
IOT	Internet of Things
FCN	Fully Convolutional Network
GCLNet	Global Style and Local Matching Contrastive Learning Network
GIS	Geographic Information Systems
LPIS	Land Parcel Identification System
MCC	Matthews correlation coefficient
ML	Machine Learning
MLP	Multilayer Perceptron
NDVI	Normalized difference vegetation index
OBIA	Object-based Image Analysis
PU	Positive Unlabeled
RF	Random Forests
R-CNN	Region-based Convolutional Neural Network
R2U-Net	Recurrent Residual U-Net
RNN	Recurrent Neural Networks
RSI	Remote Sensing Image

SGD	Stochastic Gradient Descent
SL	Supervised Learning
SSL	Self-supervised Learning
SVM	Support Vector Machines
VHR	Very High Resolution

List of symbols

α_t	Margin between positive and negative pair
α_r	Learning rate
$\frac{\partial L}{\partial W^{(l)}}$	Partial derivative of the loss function
$a^{(l)}$	Output (activation) of layer l
$a^{(l-1)}$	Input (activation) from the previous layer
$b^{(l)}$	Bias vector associated with layer l
Acc	Accuracy
$d(a, n)$	Distance between the anchor sample and the negative sample
$d(a, p)$	Distance between the anchor sample and the positive sample
$F1$	F1 score
$G(x, y)$	Gaussian function at coordinates (x, y)
I	Input image
K	Convolution kernel
L_{triplet}	Triplet loss function
\mathcal{L}_{NCE}	NCE loss function
$s(\mathbf{z}_i, \mathbf{z}_j)$	Similarity between the embeddings z_i and z_j
$\sum_{k=1}^K$	Summation over all K negative samples
$\max(0, \dots)$	Hinge loss function
$(I * K)(i, j)$	Convolution operation between image I and kernel K at position (i, j)
$\sum_m \sum_n$	Sum over all possible positions of the kernel in the input image
$I(m, n)$	Pixel value of the input image at position (m, n)
$K(i - m, j - n)$	Convolution kernel at the position $(i - m, j - n)$ to its center
MCC	Matthew's Correlation Coefficient

σ	Standard deviation of the Gaussian distribution
σ_a	Activation function applied to the weighted sum of inputs
$S(i, j)$	Convolution operation at position (i, j)
TP	True positives
TN	True negatives
FP	False positives
FN	False negatives
$W^{(l)}$	Weight matrix associated with layer l
τ	Contrastive learning temperature

Contents

1	INTRODUCTION	25
1.1	Objectives and contributions	27
1.2	Organization	28
2	THEORETICAL FOUNDATIONS	29
2.1	Geospatial image representation	29
2.1.1	Square	29
2.1.2	Triangular	31
2.1.3	Hexagonal	32
2.2	Computer vision-based methods	33
2.2.1	Region-based	33
2.2.2	Edge-based	35
2.2.3	Hybrid	36
2.3	Machine learning-based methods	36
2.3.1	Positive Unlabeled (PU)	39
2.3.2	Deep learning	40
2.3.2.1	Convolutional neural networks	40
2.3.2.2	Triplet loss siamese networks	41
2.3.2.3	Contrastive learning	42
2.4	Segmentation	44
2.4.1	Semantic segmentation	45
2.4.2	Instance segmentation	46
2.4.3	Panoptic segmentation	47
2.5	Performance measurements	48
3	CROP FIELDS DETECTION	51
3.1	Classical computer vision	51
3.2	Machine learning	52
3.3	Deep learning	53
3.4	Final considerations	56
3.5	Related work summary	57
3.6	Challenges	58
4	PRODUCTIVE CROP FIELDS DATASET	61
4.1	Dataset	61
4.2	Data processing	66

5	BENCHMARK RESULTS	69
5.1	Experimental protocol	69
5.1.1	Positive Unlabeled (PU) Learning	69
5.1.2	Triplet Loss Siamese Network	70
5.1.3	Contrastive Learning	71
5.1.4	Metrics	72
5.2	Results	73
5.2.1	PU Learning	73
5.2.2	Triplet Loss Siamese Networks	75
5.2.3	Contrastive Learning	75
5.2.4	Overall results	77
6	CONCLUSION	79
	Bibliography	83
	APPENDIX A – CROSS VALIDATION ITERATIONS	91

1 Introduction

Food production needs to grow by 70% to meet the demands of the expected world population by 2050 (NELSON et al., 2010). Motivated by this challenge, agriculture has adopted technologies to improve and optimize input returns while preserving natural resources. Integrating these technologies promotes a farming management concept known as *precision agriculture* (ZHANG; WANG; WANG, 2002). The main goal of precision agriculture is to provide tools for allowing the farmer to observe, measure, and respond to field variability in crops, facilitating faster and better decisions. In addition, these techniques are generic enough to be applied to various crops, including but not limited to corn, soy, coffee, sugarcane, beans, and even pastures (MULLA, 2013; BHAKTA; PHADIKAR; MAJUMDER, 2019).

To efficiently organize and manage large crops, farmers leverage remote sensing to divide their land into smaller observation units that this work will refer to as *agricultural* or *crop fields*. Figure 1 highlights two samples of crop fields overlaid on a satellite image. A field shape is designed based on topography and mechanization planning. For example, fields are built around contour farming and the ideal length to fill the wagon capacity in a sugarcane crop. The same logic is followed for grains in the Brazilian South and Southeast. Nonetheless, the most important variable is the harvester capacity in the Midwest, where the topography is often plain (SPEKKEN; MOLIN; ROMANELLI, 2015; GRIFFEL et al., 2019; BOLFE et al., 2020). A *productive crop field* is an area consistently used for the cycle of growth and harvest of a crop, typically yearly but more often in some regions with favorable soil and weather.



Figure 1 – Two crop fields highlighted in a satellite image. Source: own authorship.

Thanks to the availability of a massive amount of labeled data, the development of artificial intelligence (AI), particularly machine learning (ML) and deep learning (DL), has allowed for acceleration and improvement in many areas of agriculture (GARCIA-

PEDRERO et al., 2019; PERSELLO et al., 2019; MASOUD; PERSELLO; TOLPEKIN, 2020; WALDNER; DIAKOGIANNIS, 2020; WALDNER et al., 2021). Despite all the advances, having an adequate amount of labeled data to train such methods can be costly and not always affordable for precision farming.

Usually, farmers rely on experts to build the field boundaries using dedicated Geographic Information Systems (GIS) software. Despite the expert’s ability and knowledge, finding crop fields and drawing their boundaries has been a major, expensive, and time-consuming challenge (WAGNER; OPPELT, 2020b). The larger the customer’s land is, the more significant the number of fields to be created. For instance, there are farms in Brazil with more than eighteen thousand fields that require regular updates to reflect their accurate states. Finding productive fields is even more challenging because it requires previous knowledge of these areas or the analysis of satellite images over time.

Aiming to overcome these challenges, we introduce a new framework to automatically detect productive crop fields. This framework relies on agricultural machines’ operational data, i.e., the information produced by the sensors installed in these machines, to determine if the operation corresponds to a productive crop field activity. These sensors, connected to embedded controllers, can accurately inform the system if the machine is in a tillage, seeding, or harvesting process. This data is combined with Sentinel-2 images collected to build the dataset. In addition, we provide three different methods specially selected and tailored for this dataset and its particularities. The complete framework is demonstrated in Figure 2.

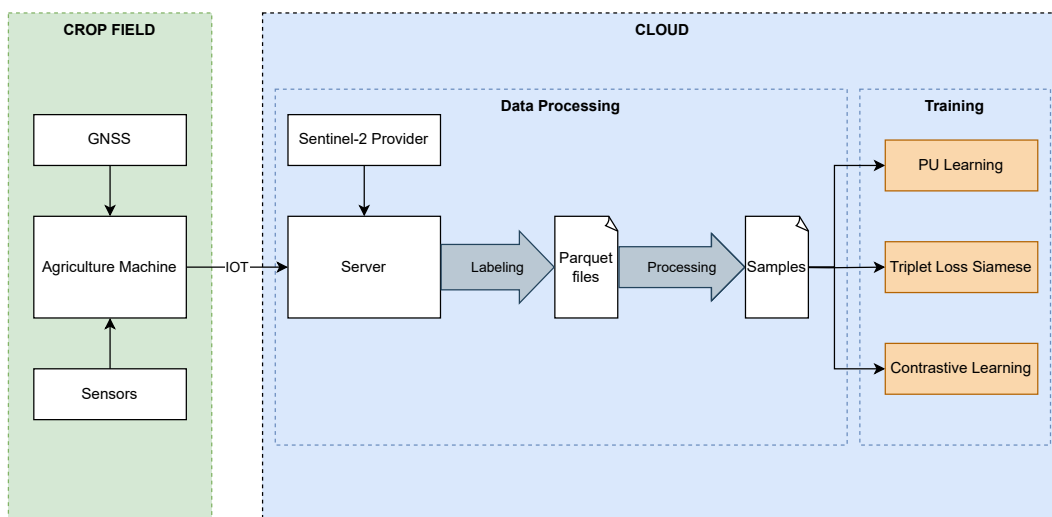


Figure 2 – End-to-end diagram of the proposed framework. Source: own authorship.

The proposed framework offers innovative approaches that differentiate it from other studies in this field.

Firstly, this information is aggregated in geospatial L12 hexagons, establishing a common ground for employing different satellite sources with different resolutions. This

means that we can easily integrate Sentinel-2 with Landsat or even aerial images with minimal effort, no source code changes or method adaptations.

Secondly, typical farmers are usually interested in monitoring productive areas rather than all possible fields since many are not fertile or suitable for any crop. For this purpose, the proposed dataset contains highly accurate positive samples (i.e., productive crop fields) and inferred negative ones, making it well-suited for positive and unlabeled learning (DENIS; GILLERON; LETOUZEY, 2005).

Finally, we offer modified versions of state-of-the-art machine learning and deep learning methods, with the respective benchmark results for further comparison. The employed methods comprehend different training strategies (i.e., supervised learning, semi-supervised, and self-supervised learning classification of unlabeled data). The latter, specifically included since recent advances suggest that self-supervised methods (e.g., contrastive learning) may also provide a promising alternative even for cases where labeled samples are scarce (GULDENRING; NALPANTIDIS, 2021).

1.1 Objectives and contributions

The main objective of this dissertation is to introduce a new framework for automatic productive crop field detection, enabling the segmentation of crop fields in satellite imagery and delineating their boundaries. This work presents the following key contributions:

- Development of a mechanism to build datasets containing pertinent information that can be employed to train machine learning models.
- Publication of the generated dataset to the scientific community to foster further studies and advancements in the area.
- Introduction of a new satellite image segmentation method tailored towards productive crop fields rather than generic ones.
- Adaptation of state-of-the-art machine learning methods, originally designed for other applications, to be applied to the proposed dataset.
- Presentation of the performance of benchmark machine learning methods that closely align with the application and the proposed dataset.
- All steps of the proposed pipeline were carefully designed to address the challenge in a pragmatic and extendable way to ease future work.

1.2 Organization

This dissertation is organized as follows:

1. In Chapter 1.2, we explore the fundamentals of image segmentation, particularly emphasizing its application in remote sensing. Additionally, we present tools and methodologies commonly associated with this field. The foundation established in this chapter is an important support for the subsequent sections.
2. In Chapter 3, we conduct a comprehensive literature review of prior research focused on crop field detection and boundary delineation across various applications. Throughout this review, we analyze the strengths and weaknesses of each study, comparing the most important points with the present dissertation and emphasizing the contributions to this domain.
3. In Chapter 4, we describe the dataset proposed within the scope of this study, detailing its development and the subsequent processing steps essential for its employment as input for machine learning methods. Concluding this chapter, we present the experimental protocol, which will serve as a foundational framework for the subsequent section.
4. In Chapter 5, we present the results obtained by applying the selected methods. We bring the merit and limitations of each approach, providing insight into their respective performance metrics to provide a clear understanding of where each method excelled and where improvements could be made.
5. Finally, this dissertation concludes by presenting a summary of findings, a final analysis, and forward-looking guidelines for future research in this domain.

2 Theoretical Foundations

In remote sensing, image segmentation takes center stage, serving as a bridge between raw data and meaningful insights encompassing tasks like land cover and land-use classification, demanding increasingly advanced image processing algorithms. Traditional digital image processing methods predominantly investigate remote sensing data pixel-by-pixel. Nevertheless, these techniques frequently encounter challenges in extracting the desired information, particularly from high-spatial resolution remote sensor data (WANG; JENSEN; IM, 2010). In connection with this matter, this chapter details some basic concepts of geospatial imagery, detailing the nuances of image representation, exploring diverse segmentation methodologies, and concluding with the machine learning methods with potential application to this area.

2.1 Geospatial image representation

In spatial analysis, using point or polygonal grids is a prevalent practice for sampling, indexing, or partitioning geographic areas (SAHR, 2011). This approach proves particularly valuable in diverse scenarios, such as overlaying a study area with a grid of points as part of a systematic spatial sampling strategy, segmenting large regions into manageable units for indexing, as exemplified by UTM grid zones, or subdividing a study area into subunits to summarize spatial variables. In the latter case, a widely adopted methodology involves employing a raster format, wherein a grid comprising uniform cells is associated with the study area. Each cell within this grid is then assigned a value corresponding to the spatial variables of interest. In fields like ecology and conservation, these variables may include, for example, counts of individuals of a threatened species per grid cell, elevation, mean annual rainfall, or land use classifications. The most commonly used grids are square, triangular and hexagonal.

This section briefly describes these representations, detailing the merits and limitations associated with each.

2.1.1 Square

The prevalent use of square grids finds its most common applications in raster datasets, geohashes (SUWARDI et al., 2015) and quadkeys. To start, the pixels of a satellite image in raster format can be recognized as the finest version of a square grid system. On a more practical approach, a square grid system can be implemented by utilizing geohashes. A geohash represents a hierarchical data structure that converts 2D spatial points, defined by latitude and longitude, into concise alphanumeric strings. Operating on a global scale,

geohashes intricately divide the world into a grid comprising 32 cells, organized into 4 rows and 8 columns, as shown in the Figure 3.

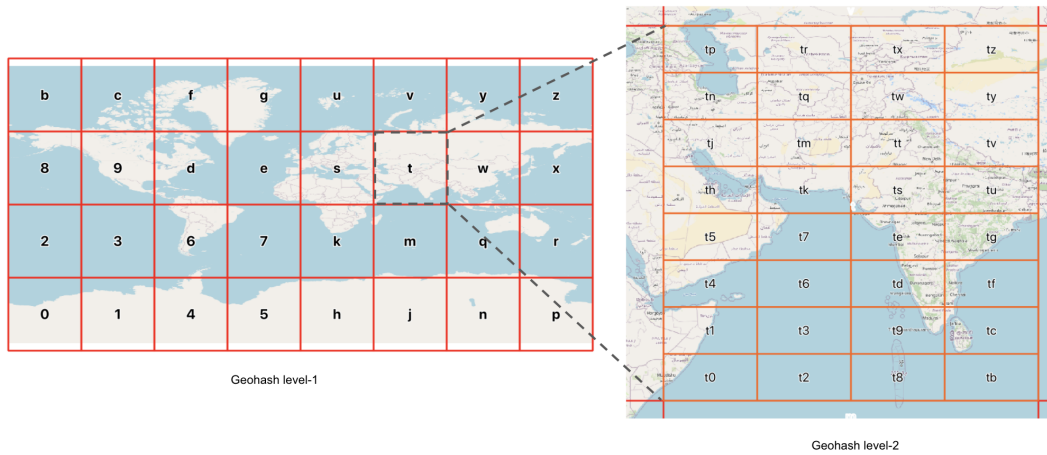


Figure 3 – Geohash grid system and leveling. Source: Geospatial World website¹.

The intuition of geohashes lies in their recursive nature; each cell within the grid can be subdivided into an additional grid of 32 cells. Consequently, the length of the geohash string directly correlates with the level of spatial accuracy achieved. Consequently, the hierarchical structure facilitates the identification of spatial proximity: geohashes sharing a common prefix indicate spatial closeness. Therefore, the longer the common prefix, the closer the associated locations. The main usages for geohashes are providing a unique identifier for the associated regions and representing a point in a database.

As a noteworthy alternative, Microsoft developed a square grid indexing system concept that composed the Bing Maps Tile System², most commonly known by *quadkeys*. The term quadkey is an abbreviation for *quadtree key*. These keys encode square regions within the latitude and longitude space, organized according to different detail levels. At the initial level, the Earth’s entire surface divides into four quadkeys, similar to a map’s zoom level that provides a panoramic view of the entire world. Each quadkey is associated with a single-digit code ranging from zero to three. Upon zooming into the next level, the original four quadkeys are divided into four, with an additional digit appended to the code. The visual representation in the Figure 4 illustrates this progressive refinement process.

As previously exemplified, implementing a square grid system can be applied through different alternatives. However, while recognized as a straightforward method, it has several drawbacks. Firstly, it introduces substantial size distortion, leading to significant

¹ Polygeohasher: an optimized way to create geohashes is available at <<https://www.geospatialworld.net/blogs/polygeohasher-an-optimized-way-to-create-geohashes/>>. Access on December 10, 2023.

¹ Bing Maps Tile System is available at <<https://learn.microsoft.com/en-us/bingmaps/articles/bing-maps-tile-system>>. Access on December 10, 2023.

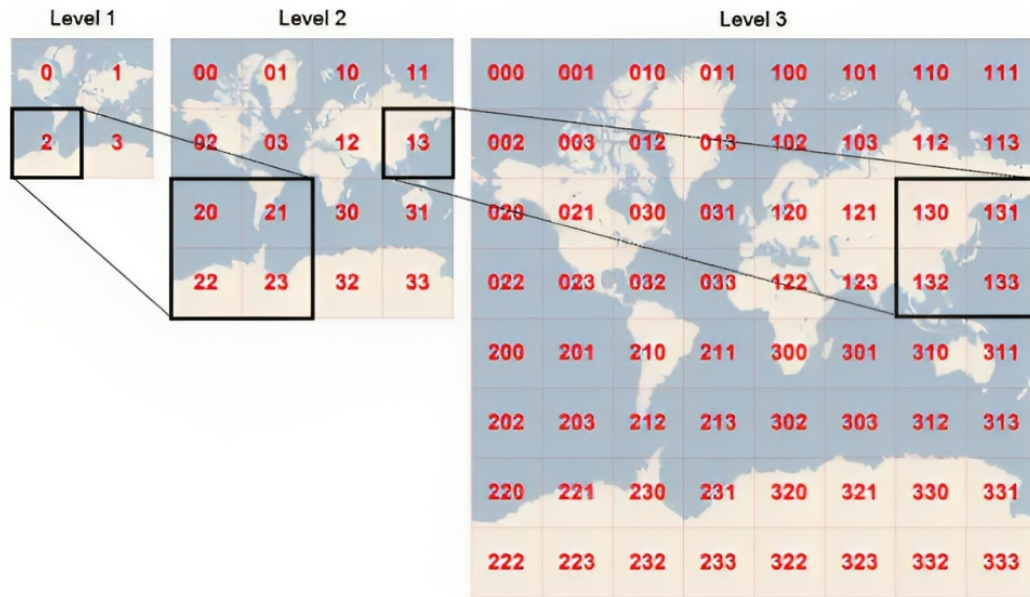


Figure 4 – Quadkey codes according to location and detail level. Source: Microsoft Bing website.

variations in the areas of different cells along the globe. Additionally, square grids present challenges, demanding multiple sets of coefficients for analytical purposes. This challenge arises from squares having two distinct types of neighbors: one type sharing an edge in the four cardinal directions, and another sharing a vertex in four diagonal directions (SAHR, 2011). In the next sections, we explore alternative grid systems utilizing different shape formats, aiming to minimize the limitations exposed here.

2.1.2 Triangular

Triangular grids, sometimes associated in pairs to achieve diamond shapes, remain relatively uncommon in practical applications. This rarity can be attributed to a combination of factors, including their unfamiliarity and specific geometric properties that pose challenges for seamless integration on maps (KIMERLING KEVIN SAHR; SONG, 1999).

One key obstacle lies in the inherent geometric characteristics of triangles, specifically their tendency to have a larger perimeter relative to their enclosed area. This unfavorable ratio makes assembling triangular grids cohesively on a map more challenging, contributing to their infrequent use in spatial representations.

Additionally, the limited connectivity of triangular grids imposes an additional obstacle. Each triangle is linked to only three adjacent triangles, constraining movement options and connections. This constrained connectivity contrasts with the more versatile arrangements facilitated by square and hexagonal grids, limiting the adaptability of triangular grids in diverse mapping scenarios.

Furthermore, the asymmetry inherent in triangular grids further complicates their

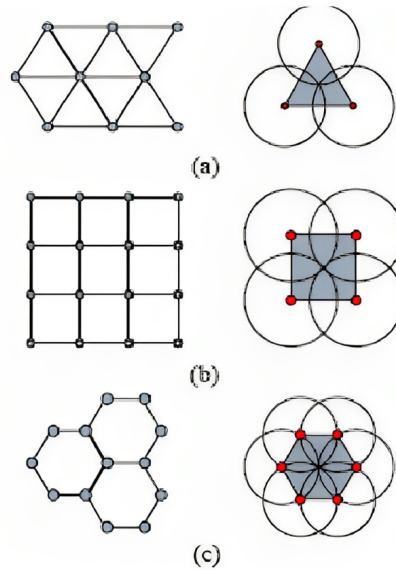


Figure 5 – Overlapping of radio-based systems using a triangular grid system. Source: [Kumari e Singh \(2012\)](#).

adoption. Unlike hexagons and squares, where two faces are parallel, triangles introduce two directions along which lines are both centered from the axis of movement. This lack of symmetry adds a layer of complexity to the interpretation and utilization of triangular grids in spatial contexts.

Nevertheless, even with all these considerations, there are still scenarios where a triangular grid system excels, for example, in applications involving the planning of radio communication systems. In this case, the radiofrequency overlapping is reduced using this specific system, as demonstrated in Figure 5 ([KUMARI; SINGH, 2012](#)), making it an appropriate choice for a grid system.

2.1.3 Hexagonal

Beyond their aesthetic appeal, hexagons offer a high degree of symmetry, surpassing the symmetry of geohashes and closely resembling circles in shape, enabling more accurate spatial sampling. This geometric advantage has encouraged widespread adoption, notably by companies such as Uber, recognizing the efficacy of hexagons in various spatial applications ([YUE et al., 2021](#)).

An insight into hexagons reveals their similarity with triangular grids, a geometric relationship where placing a dot in the center of each hexagon and connecting them to adjacent ones results in a triangular grid, as shown in the Figure 6 ([NAGY, 2022](#)). This characteristic highlights the versatility of hexagons, particularly in comparison to triangular and square grids. ([HERRMANN; KAMPHANS; LANGETEPE, 2010](#)).

The choice between hexagons and geohashes depends on specific use cases, each necessitating trade-offs. Analyzing parameters such as distance from the nearest cells,

hexagons exhibit an advantageous property. They are equidistant to their neighbors, simplifying analysis and rendering them preferable for tasks involving connectivity or movement. Furthermore, hexagons emerge as the optimal choice for fitting on curved surfaces, offering a dense tessellation that minimizes edge effects and accommodates the curvature of the Earth more effectively than squares or triangles. And that is the primary reason why most recent geospatial applications have adopted hexagonal grids (YUE et al., 2021).

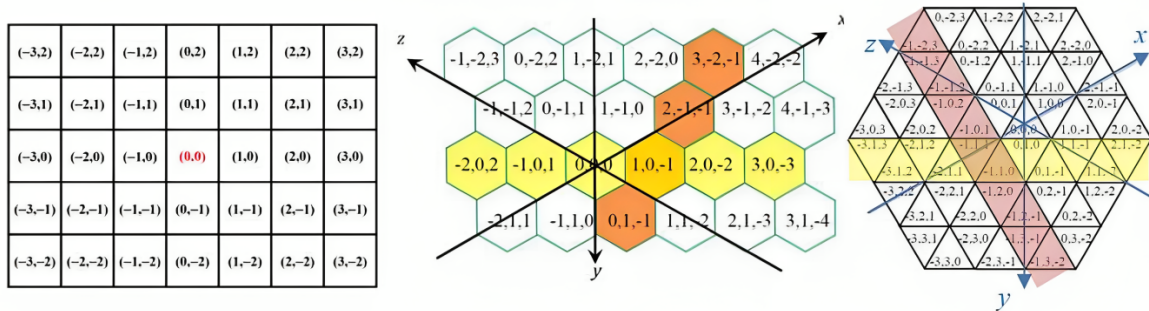


Figure 6 – The square, hexagonal and triangular grids with their symmetric coordinate systems. Source: Nagy (2022).

Hexagons also excel in revealing explicit patterns in data, especially when compared to linear figures like squares and rectangles. The inherent shape of hexagons allows for the easy and explicit display of curvature in data patterns, avoiding the visual impediment posed by straight, unbroken lines inherent to square and rectangular shapes. This versatility and efficiency in addressing various spatial considerations establish hexagons as a powerful visualization and spatial analysis tool. The methods explored in the following sections can perform the latter individually or in agglomerates.

2.2 Computer vision-based methods

In this section, we introduce the fundamentals of classical computer vision methods for image segmentation. These techniques have played a crucial role in resolving the complexities inherent in visual data analysis. Our investigation extends to an examination of classical approaches, covering region-based, edge-based, and hybrid methodologies as well as their strengths and limitations.

2.2.1 Region-based

Region-based segmentation is a crucial technique in computer vision that involves partitioning an image into regions or segments based on certain criteria such as color, texture, or intensity. Unlike pixel-based methods, which classify individual pixels independently, region-based segmentation groups pixels with similar properties into coherent

regions. This approach aims to capture higher-level information and structures within an image, facilitating more meaningful interpretation and analysis.

One well-known region-based segmentation method is the watershed algorithm (BEUCHER; MEYER, 1993). The watershed algorithm treats an image as a topographical surface, where basins represent regions, and the flooding of these basins corresponds to the segmentation process. This process is demonstrated in the Figure 7. By leveraging morphological operations, the watershed algorithm is particularly effective in segmenting images with varying intensities and gradients.

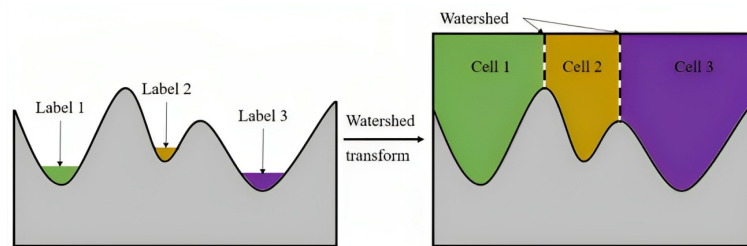


Figure 7 – The segmentation process of the watershed algorithm. Source: Zheng et al. (2021).

Another influential region-based segmentation technique is the GrabCut algorithm (ROTHER; KOLMOGOROV; BLAKE, 2004). GrabCut formulates segmentation as an energy minimization problem and utilizes graph cuts for interactive foreground extraction. By iteratively updating the segmentation based on user input, GrabCut achieves accurate and interactive segmentation, making it valuable for applications such as image editing and object extraction.

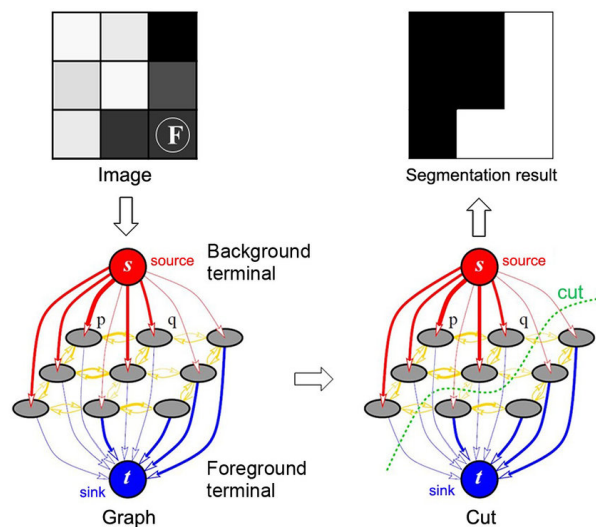


Figure 8 – Simplified example of a GrabCut segmentation process. Source: Xiao et al. (2017).

In the Figure 8, this process is demonstrated in a segmentation for a simple 3×3 image. The pixel labeled “F” is a hard constraint, and the image feature term defines the

weights of the n -links (yellow edges), while the change feature term defines the weights of the t -links (red and blue edges). Each edge's weight, or cost, is reflected by its thickness.

These region-based segmentation methods, among many others using classical computer vision, still find usage in many applications, including medical imaging and object recognition, especially where there are computational capability constraints.

2.2.2 Edge-based

Edge-based computer vision segmentation is another classical approach that relies on detecting and emphasizing boundaries, or edges, within an image to distinguish between different regions. The primary idea is that abrupt changes in intensity or color often correspond to object boundaries, making edges a key feature for segmentation. Edge-based methods are foundational in computer vision and widely used in applications such as object recognition, image analysis, and medical imaging (CANNY, 1986).

One of the fundamental techniques in edge-based segmentation is the Canny edge detector (CANNY, 1986). The Canny edge detector identifies edges by locating points where the gradient of the image intensity undergoes a significant change. This method is characterized by its ability to detect edges accurately while suppressing noise and providing thin and well-connected contours.

The edge detection process often involves the computation of gradients, typically using convolution with gradient masks and subsequent application of a threshold to identify significant edges. The Canny edge detector can be mathematically represented as in the Equation 2.1. In the Canny edge detection algorithm, the value of σ affects the scale of the smoothing operation, and the choice of σ depends on the characteristics of the image and the desired level of smoothing. Commonly, σ is chosen based on experimentation and the specific requirements of the application.

$$G(x, y) = \frac{1}{\sqrt{2\pi}\sigma} e^{-(x^2+y^2)/(2\sigma^2)} \quad (2.1)$$

Another widely used edge detection operator is the Sobel operator, which approximates image gradients using convolution with Sobel kernels (NAGENDRA et al., 1993). The Sobel operator is computationally efficient and is often employed as a preprocessing step in edge-based segmentation methods.

Edge-based segmentation can be refined by incorporating additional techniques such as edge linking and contour extraction. The extraction of contours helps in delineating object boundaries more precisely.

To conclude, computational efficiency and robustness to illumination changes contribute to edge detection effectiveness in scenarios where clear borders delineate objects.

However, these methods may struggle in the presence of noise and can be less discriminative in cases where objects share similar colors or textures. On the other hand, region-based segmentation techniques, such as the aforementioned graph-cut methods, focus on grouping pixels based on overall similarity, emphasizing homogeneity within segments. This makes them more versatile, particularly in handling images with gradual intensity changes or complex textures. As an advancement, hybrid methods combining the best of these two approaches were introduced.

2.2.3 Hybrid

Hybrid segmentation methods integrate both region-based and edge-based techniques to exploit the complementary strengths of each approach, aiming to achieve more accurate and robust segmentation results. These methods acknowledge that edge information provides crucial boundary details, while region-based methods offer a more global context for segmentation. Combining these strategies, hybrid segmentation methods aim to overcome limitations inherent in individual approaches.

One common strategy in hybrid segmentation is to use edge information as a feature or constraint within region-based methods. For instance, incorporating edge data into graph-cut segmentation can enhance the precision of object boundaries. The edges serve as additional cues, guiding the segmentation process to align with the more detailed information provided by the boundary emphasis of edge-based techniques. Mathematically, based on the theory explored in the previous sections, this integration can be expressed as an extended energy function, where the data term considers both region homogeneity and edge information (RYDBERG; BORGEFORS, 2001).

2.3 Machine learning-based methods

Machine learning is a subfield of artificial intelligence (AI) that focuses on developing algorithms and models capable of learning from data to make predictions or decisions without explicit programming. The fundamental idea behind machine learning is to enable computers to automatically learn and improve their performance on a specific task through experience (or training) without being explicitly programmed for that task. This paradigm shift from traditional if-based programming to data-driven learning has enabled machines to perform complex tasks such as image and speech recognition, natural language processing, and autonomous decision-making (BISHOP, 2006).

One of the foundational concepts in machine learning is the use of training data to train models, where algorithms learn patterns and relationships from the data to generalize and make predictions on new, unforeseen data. The field encompasses a variety of techniques, including supervised learning, unsupervised learning, and reinforcement

learning, each serving different purposes in solving diverse problems. Machine learning has witnessed rapid advancements in recent years, fueled by the availability of large datasets, improved computational power, and innovative algorithms (GOODFELLOW; BENGIO; COURVILLE, 2016).

Machine learning has undergone a transformative evolution with the advances of neural networks, a subfield that draws inspiration from the architecture of the human brain (NIELSEN, 2015). Neural networks, also called artificial neural networks (ANNs), have emerged as a powerful framework within machine learning, particularly in solving complex problems. These networks consist of interconnected layers of nodes, known as neurons, where each connection has an associated weight that is adjusted during the training process. Through a series of forward and backward passes, neural networks can learn intricate patterns and representations from data, making them well-suited for tasks ranging from image and speech recognition to natural language processing.

Neural networks operate through a series of interconnected layers of artificial neurons, and their learning process involves two key phases: forward propagation and backpropagation.

In forward propagation, represented in Figure 9, the process begins with the input layer, where the neural network receives the initial data or features. Each neuron in the subsequent layers computes a weighted sum of its inputs, incorporating the weights assigned to the connections. This sum is then passed through an activation function, introducing non-linearity to the network (LECUN; BENGIO; HINTON, 2015).

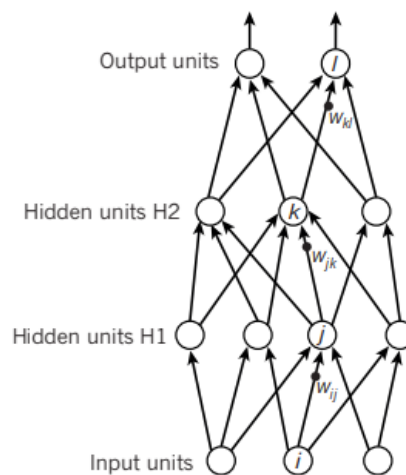


Figure 9 – Neural network basic forward propagation. Source: LeCun, Bengio e Hinton (2015).

The final layer produces the network's output, representing the predicted values or classifications. The forward propagation step can be mathematically expressed as in the Equation 2.2.

$$a^{(l)} = \sigma_a \left(W^{(l)} \cdot a^{(l-1)} + b^{(l)} \right) \quad (2.2)$$

In sequence, in backpropagation, the process starts with the output of the forward propagation being compared to the actual target values, and a loss is calculated. The backpropagation process is shown in Figure 10.

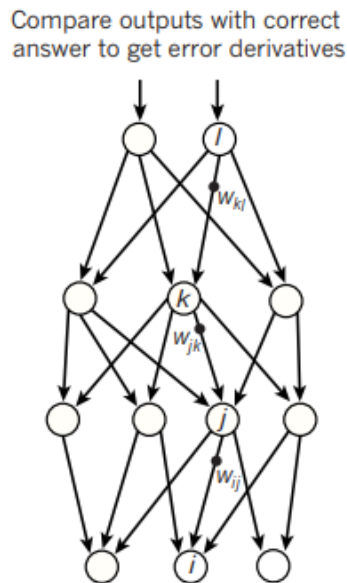


Figure 10 – Neural network backpropagation. Source: [LeCun, Bengio e Hinton \(2015\)](#).

The loss function quantifies the disparity between the predicted and actual values. Backpropagation involves propagating this loss backward through the network to update the weights and biases, represented in the Equations 2.3 and 2.4, respectively.

$$W^{(l)} \leftarrow W^{(l)} - \alpha_r \frac{\partial L}{\partial W^{(l)}} \quad (2.3)$$

$$b^{(l)} \leftarrow b^{(l)} - \alpha_r \frac{\partial L}{\partial b^{(l)}} \quad (2.4)$$

This is achieved through the chain rule of calculus, calculating the gradients of the loss with respect to the weights and biases. The gradients are then used to update the weights and biases in the network, iteratively minimizing the loss. This step is usually performed using optimization algorithms like stochastic gradient descent (SGD) or its variants ([LECUN; BENGIO; HINTON, 2015](#)). Mathematically, the backpropagation step involves computing the $\frac{\partial L}{\partial W}$ and $\frac{\partial L}{\partial b}$ gradients for each layer, where L is the loss function. The weights and biases are then updated using these gradients.

The iterative execution of forward and backward propagation during training allows the neural network to learn to optimize weights and biases that minimize the prediction

error on the training data. This adaptive learning process enables neural networks to generalize well to unseen data and perform effectively in various tasks.

A distinctive feature of neural networks is their ability to learn hierarchical representations of data, allowing them to capture intricate features and relationships. This characteristic has paved the way for developing deep neural networks, which are neural networks with multiple layers (LECUN; BENGIO; HINTON, 2015).

2.3.1 Positive Unlabeled (PU)

It is a semi-supervised classification approach of unlabeled data, particularly suitable for the current study, given the availability of accurate positive data, the lack of negative ones, and the ease of obtaining unlabeled data. PU learning (DENIS; GILLERON; LETOUZEY, 2005) relies on supervised methods, and we selected Support Vector Machines (SVM) for this evaluation as well as Random Forests (RF) and Multilayer Perceptron (MLP) networks.

PU learning differs from One-Class learning (KHAN; MADDEN, 2010) by having, besides positive labeled samples, a group of samples whose label is unknown. In the One-Class classification, different from our scenario, the model would be trained only with positive samples, for example, and be exposed to other classes only during classification.

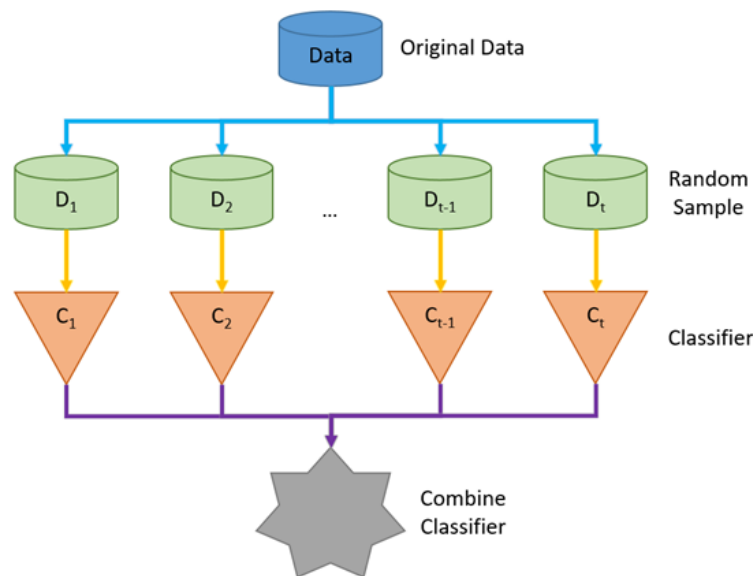


Figure 11 – Parallel bagging used in PU learning. Source: Denis, Gilleron e Letouzey (2005).

The approach relies on bootstrap aggregating (bagging) methods (BREIMAN, 1996), wherein the algorithm systematically trains a series of classifiers to distinguish known positive instances from randomly selected subsamples of the unlabeled dataset. It then averages their predictions into only one classifier, as demonstrated in Figure 11.

2.3.2 Deep learning

Deep learning has demonstrated remarkable success in various applications, such as computer vision, natural language processing, and speech recognition (GOODFELLOW; BENGIO; COURVILLE, 2016).

As neural networks evolve, researchers and practitioners explore innovative architectures and methodologies. Convolutional Neural Networks (CNNs) (LECUN; BENGIO; HINTON, 2015) excel in image-related tasks, Recurrent Neural Networks (RNNs) (SALEHINEJAD et al., 2017) are effective for sequential data, and Transformers (LIN et al., 2022) have shown unparalleled success in natural language processing. The versatility of neural networks, coupled with advancements in hardware and the availability of large datasets, underscores their importance in shaping the future of machine learning.

2.3.2.1 Convolutional neural networks

A Convolutional Neural Network (CNN) is a feedforward neural network designed to automatically extract features from data using convolutional structures. Unlike traditional feature extraction methods, CNNs eliminate the need for manual feature extraction. Inspired by visual perception, the architecture of CNNs mirrors biological neurons, with artificial neurons corresponding to their biological counterparts. CNN kernels are receptors responsive to various features, while activation functions simulate the neural transmission threshold. Loss functions and optimizers are integral components that guide the CNN in learning desired patterns (LECUN et al., 1998).

CNNs offer several advantages over traditional artificial neural networks. Firstly, they employ local connections, where each neuron is connected to a few neurons in the previous layer, effectively reducing parameters and accelerating convergence. Secondly, weight sharing allows a group of connections to share the same weights, further minimizing parameters. Thirdly, down-sampling through pooling layers leverages local image correlation principles, reducing data volume while preserving crucial information and eliminating trivial features. These characteristics establish CNNs as representative algorithms in deep learning (LI et al., 2022b).

Four essential components are typically employed to construct a CNN model. Convolution, a crucial step for feature extraction, generates feature maps. Padding is introduced to address information loss at the border, which is achieved by expanding the input with zero values. Stride controls the convolution density, with a larger stride resulting in lower density. Finally, pooling, such as max and average pooling, mitigates redundancy and potential overfitting issues in feature maps. These steps are demonstrated in Figure 12.

The convolution operation can be mathematically represented as in the Equation 2.5.

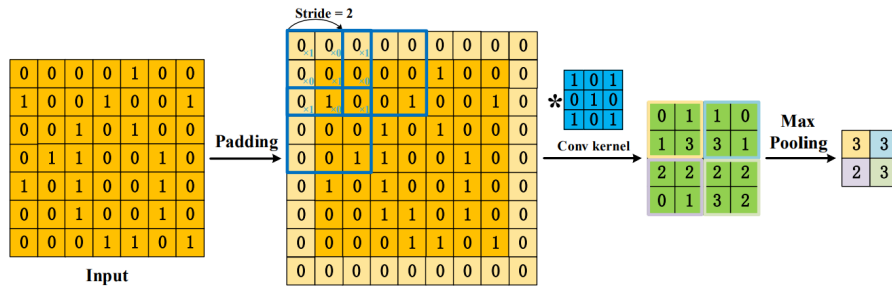


Figure 12 – Steps of a 2D Convolutional Neural Networks. Source: Li et al. (2022b).

In the equation, I is the input image, K is the convolutional kernel, and S is the output feature map. This operation is crucial for capturing spatial hierarchies in data.

$$S(i, j) = (I * K)(i, j) = \sum_m \sum_n I(m, n) \cdot K(i - m, j - n) \quad (2.5)$$

Dilated convolution was introduced to address the challenge of perceiving larger areas. Deformable convolution addresses irregular shapes in real-world objects by allowing focused attention on specific regions. This adaptability enhances the representativeness of feature maps.

To conclude, various advanced convolutional techniques contribute to the versatility and effectiveness of CNNs across diverse applications. Different architectures to address different applications will be discussed in the next sections.

2.3.2.2 Triplet loss siamese networks

Triplet Loss Siamese Networks, a popular approach in deep learning, are designed for face recognition, image retrieval, and similarity learning (SCHROFF; KALENICHENKO; PHILBIN, 2015). These networks are particularly effective in learning embeddings that map input instances into a high-dimensional space, where the distance between embeddings reflects the similarity or dissimilarity between the instances. A key advantage of Triplet Loss Siamese Networks lies in their ability to learn discriminative features by training on triplets of examples: an anchor instance, a positive instance (similar to the anchor), and a negative instance (dissimilar to the anchor). This basic concept is shown in the Figure 13.

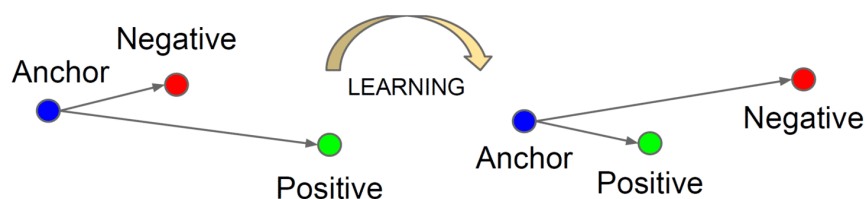


Figure 13 – Triplet loss learning process. Source: Schroff, Kalenichenko e Philbin (2015).

The fundamental idea behind Triplet Loss Siamese Networks is to minimize the distance between the anchor and positive instances, while maximizing the distance between the anchor and negative instances. This is achieved by using a triplet loss function, typically defined as in the Equation 2.6, where $d(x, y)$ represents the distance between instances and α is a margin that enforces a minimum separation between positive and negative pairs.

$$L_{\text{triplet}} = \max(0, d(a, n) - d(a, p) + \alpha_t) \quad (2.6)$$

One pioneering work in Triplet Loss Siamese Networks is the FaceNet (SCHROFF; KALENICHENKO; PHILBIN, 2015), which introduced the use of triplets to learn discriminative embeddings for face recognition. Since then, various modifications and improvements have been proposed, adapting the triplet loss for different applications and addressing challenges such as data imbalance and selecting informative triplets.

The Triplet Loss Siamese Network architecture has proven to be a robust and effective approach for learning embeddings that capture intricate relationships between instances. It is a valuable tool in similarity learning and feature representation.

2.3.2.3 Contrastive learning

Contrastive learning has emerged as a robust technique, enabling models to harness extensive amounts of unlabeled data and enhance performance even when labeled data is scarce. The core of contrastive learning directs to incentivizing the mapping of similar instances closer together in a learned embedding space while pushing dissimilar instances further apart. By executing the learning process as a discrimination task, contrastive learning enables models to distinguish features and similarities in the data (LE-KHAC; HEALY; SMEATON, 2020).

The journey of contrastive learning typically begins with data augmentation, a process involving the application of various transformations or perturbations to unlabeled data to generate diverse instances or augmented views. Techniques such as cropping, flipping, rotation, random cropping, and color transformations inject variability into the data, exposing the model to different perspectives of the same instance, as shown in the Figure 14. This diversity ensures that the model learns to capture relevant information independent of variations in the input data (CHEN et al., 2020).

Following data augmentation, the next phase involves training an encoder network. This network takes the augmented instances as input and maps them to a latent representation space where meaningful features and similarities are captured. The encoder network, often a deep neural network architecture like a CNN for image data, extracts and encodes high-level representations from the augmented instances, facilitating discrimination between similar and dissimilar instances in subsequent steps.



Figure 14 – Data augmentation examples applied to an image. Source: [Le-Khac, Healy e Smeaton \(2020\)](#).

A projection network is employed to refine the learned representations. This network takes the output of the encoder network and projects it onto a lower-dimensional space, known as the projection or embedding space. This additional projection step enhances the discriminative power of the learned representations by reducing complexity and redundancy in the data, aiding in better separation between similar and dissimilar instances.

The contrastive learning objective comes once the augmented instances are encoded and projected into the embedding space. The goal is to maximize agreement between positive pairs (*i.e.*, instances from the same sample) and minimize agreement between negative pairs (*i.e.*, instances from different samples). This encourages the model to bring similar instances closer together while pushing dissimilar instances apart, with similarity measured by a distance metric like Euclidean distance or cosine similarity. The model is trained to minimize the distance between positive pairs and maximize the distance between negative pairs in the embedding space.

Contrastive learning employs various loss functions to define the learning objectives, crucial for guiding the model to capture meaningful representations and differentiate between similar and dissimilar instances. The choice of the appropriate loss function depends on task requirements and data characteristics, with each loss function aiming to facilitate the learning of representations that effectively capture meaningful similarities and differences within the data. One popular example is the InfoNCE, demonstrated in the Equation 2.7.

$$\mathcal{L}_{\text{NCE}} = -\log \left(\frac{\exp(s(\mathbf{z}_i, \mathbf{z}_j)/\tau)}{\sum_{k=1}^K \exp(s(\mathbf{z}_i, \mathbf{z}_k)/\tau)} \right) \quad (2.7)$$

Once the loss function is defined, the model undergoes training on a large unlabeled dataset. The iterative optimization process involves updating the model's parameters to minimize the loss function, typically using optimization algorithms. Batch-wise updates, where a subset of augmented instances is processed simultaneously, are commonly used during this training process.

While training, the model learns to capture relevant features and similarities in the data. The iterative optimization process gradually refines the learned representations, improving discrimination and separation between similar and dissimilar instances.

Finally, a common approach is to connect a final step at the end of the network, a process usually called fine-tuning, to allow the final architecture to produce results as expected by the specified application.

2.4 Segmentation

Image segmentation is a crucial aspect of various visual representation systems, involving dividing images or video frames into distinct regions or objects. Its significance expands across diverse applications, including medical image analysis, autonomous vehicles, video surveillance, and augmented reality (MINAEE et al., 2021).

Numerous image segmentation algorithms have been developed, ranging from early techniques like watershed to more advanced methods like graph cuts. In recent years, deep learning models have introduced a new era of image segmentation with impressive performance improvements. These models often achieve the highest accuracy rates on popular benchmarks, leading to a paradigm shift in the field (CHEN et al., 2018).

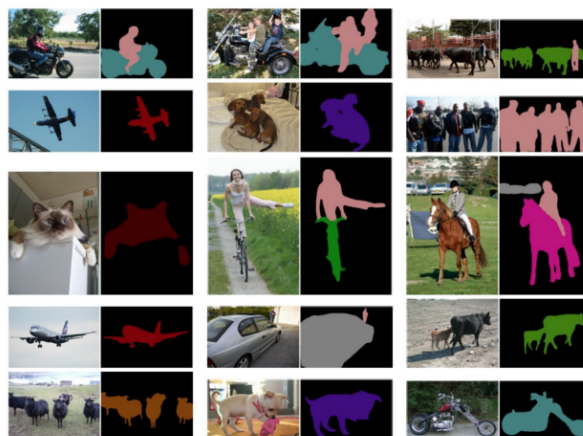


Figure 15 – Segmentation results on sample images. Source: Minaee et al. (2021).

As illustrated in Figure 15, the outputs of a widely used deep learning model exemplify the advancements in image segmentation. The task of image segmentation can be approached as a classification problem, either labeling pixels with semantic categories

(semantic segmentation), partitioning individual objects (instance segmentation), or combining the two approaches to get the best of the two worlds (panoptic segmentation). In the following sections, each of these approaches will be detailed.

2.4.1 Semantic segmentation

Semantic segmentation is a fundamental task in computer vision, aiming to classify each pixel in an image into categories. Unlike image classification, which assigns a single label to the entire image, semantic segmentation provides a detailed understanding of the spatial distribution of objects within a scene, enabling machines to perceive and interpret visual information at a pixel level (SHELHAMER; LONG; DARRELL, 2015). Figure 16 demonstrates a remote sensing image and the respective semantic segmentation result.



Figure 16 – Semantic segmentation sample with the original image on the left and segmented output on the right. Source: Carvalho et al. (2021).

One of the initial works in semantic segmentation is the Fully Convolutional Network (FCN) introduced by Shelhamer, Long e Darrell (2015). FCN employs end-to-end convolutional neural networks to process images at the pixel level, enabling real-time semantic segmentation. This marked a significant departure from traditional methods, demonstrating the efficacy of deep learning in this domain.

Benchmark datasets, such as Pascal VOC³, have been crucial in advancing semantic segmentation research. These datasets provide large-scale, annotated images for training and evaluation, allowing researchers to develop and compare models effectively.

Additionally, the U-Net architecture, proposed by Ronneberger, Fischer e Brox (2015), further contributed to the field of semantic segmentation. U-Net’s unique architecture, featuring a contracting and expansive path, facilitates the precise localization of objects in medical images and beyond, demonstrating the versatility of semantic segmentation applications.

³ Pascal Visual Object Classes is available at <http://host.robots.ox.ac.uk/pascal/VOC/>. Access on December 11, 2023.

Semantic segmentation finds application in various domains, including medical image analysis, autonomous vehicles, and scene understanding in robotics (CHEN et al., 2018). As technology advances, semantic segmentation models are continually refined, enhancing their accuracy and applicability in real-world scenarios.

2.4.2 Instance segmentation

Instance segmentation is an advancement in computer vision, surpassing traditional semantic segmentation methods by not only categorizing pixels into classes but also identifying and delineating individual instances of objects within an image. This task is crucial in applications where a detailed understanding of the distinct entities in a scene is required (HAFIZ; BHAT, 2020). Figure 17 shows a satellite image and the corresponding instance segmentation.

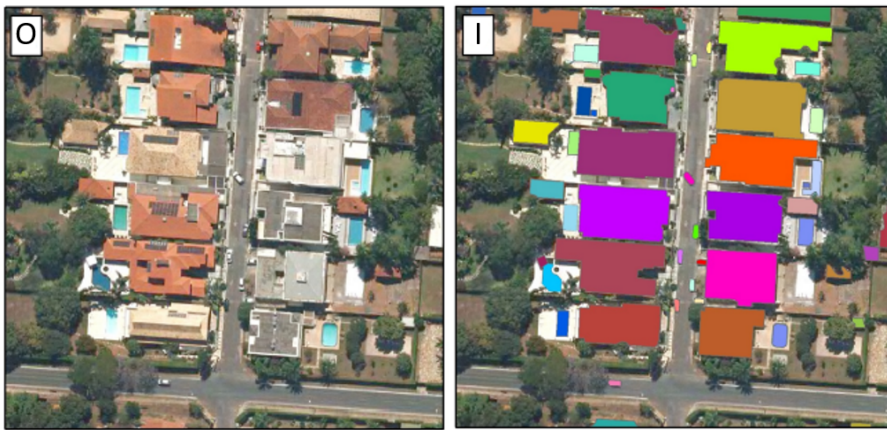


Figure 17 – Instance segmentation sample with original image on the left and segmented output on the right. Source: Carvalho et al. (2021).

One of the pioneering approaches to instance segmentation is Mask R-CNN (Region-based Convolutional Neural Network), introduced by He et al. (2017) in 2017. Mask R-CNN builds upon the success of Faster R-CNN (REN et al., 2015), incorporating an additional branch to generate segmentation masks for each detected object. This architecture has become a cornerstone for many subsequent developments in instance segmentation due to its ability to provide pixel-level accuracy.

The COCO dataset⁴ has been important in advancing instance segmentation research. With pixel-level annotations for over 200,000 images across 80 object categories, COCO provides a diverse and challenging benchmark for evaluating the performance of instance segmentation models. The dataset has fueled the development of algorithms capable of handling complex scenes with overlapping and closely packed objects.

⁴ Common Objects in Context is available at <<https://cocodataset.org/>>. Access on December 11, 2023.

In conclusion, instance segmentation is crucial to some specific applications, enabling precise identification and delineation of individual objects within an image. The continual evolution of models and datasets, such as Mask R-CNN and COCO, highlights the importance of instance segmentation in advancing our ability to comprehend and interact with visual information.

2.4.3 Panoptic segmentation

Panoptic segmentation represents a significant advance in the segmentation field, combining the strengths of semantic and instance segmentation to provide a comprehensive understanding of visual scenes. Unlike traditional segmentation methods, panoptic segmentation aims to label every pixel in an image with a semantic class or identify individual instances of objects, offering a unified approach to scene understanding [Chuang, Zhang e Zhao \(2023\)](#). Figure 18 demonstrates a satellite image and the corresponding panoptic segmentation.

The Panoptic Segmentation task was introduced by [Kirillov et al. \(2019\)](#), defining a framework that, as previously mentioned, unifies semantic and instance segmentation tasks. In panoptic segmentation, each pixel is assigned a semantic label for non-countable entities like sky or road or is associated with an instance label for countable objects like people or cars. This approach has become a benchmark in assessing the capabilities of computer vision models [Chuang, Zhang e Zhao \(2023\)](#).



Figure 18 – Panoptic segmentation example. The original image is on the left, and the segmented output is on the right. Source: [Carvalho et al. \(2021\)](#).

Efforts to advance panoptic segmentation have also led to the creation of benchmark datasets, with the Mapillary Vistas⁵ dataset being one notable example. Mapillary Vistas provides a diverse collection of street-level images with pixel-level annotations for semantic

⁵ Mapillary Vistas Dataset is available at <https://www.mapillary.com/>. Access on December 11, 2023.

and instance segmentation, fostering the development of algorithms capable of handling complex urban scenes.

Recent research has explored panoptic segmentation’s applications in real-world scenarios, ranging from urban planning and autonomous navigation to remote sensing (CARVALHO et al., 2021). The versatility of panoptic segmentation lies in its ability to simultaneously provide a high-level understanding of scene semantics and precise identification of individual objects.

Finally, panoptic segmentation represents a pivotal step towards a more comprehensive understanding of visual scenes. The framework introduced by Kirillov et al. (2019) and the availability of datasets like Mapillary Vistas have catalyzed research, resulting in the development of models capable of seamlessly integrating semantic and instance segmentation tasks.

2.5 Performance measurements

Assessing the efficacy of image segmentation methods depends on employing robust performance measurement metrics that provide insights into their accuracy and reliability. Precision, recall, F1 score, and Matthews correlation coefficient (MCC) are key indicators. Collectively, these metrics offer a comprehensive evaluation of a segmentation algorithm’s performance by evaluating its ability to achieve accurate and balanced results.

Accuracy, a fundamental metric shown in Equation 2.8, gauges the overall correctness of a segmentation method by measuring the ratio of correctly predicted pixels to the total number of pixels. While accuracy provides a broad overview, it may not be sufficient in scenarios where imbalanced datasets or specific misclassification types are critical considerations.

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (2.8)$$

In sequence, the F1 Score, a harmonic mean of precision and recall, emerges as a pivotal metric that balances false positives and false negatives. Its equation is shown in Equation 2.9. It comprehensively captures both the positive predictive value and sensitivity, making it particularly valuable when achieving a harmonious mix of precision and recall is important.

$$\text{F1} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2 \times \text{TP}}{2 \times \text{TP} + \text{FP} + \text{FN}} \quad (2.9)$$

To overcome the challenges posed by some applications, Matthews Correlation Coefficient (MCC), demonstrated in Equation 2.10, assesses the correlation between

predicted and true positive instances while accounting for potential imbalances in the dataset. MCC ranges from -1 to 1, providing a robust measure of a segmentation method's ability to handle both positive and negative instances.

$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN})}} \quad (2.10)$$

In image segmentation, where accurately delineating objects or structures is crucial, these metrics play an important role. By prioritizing the combined assessment of precision, recall, F1 score, and MCC, researchers and practitioners gain a more nuanced understanding of the performance of segmentation methods. This approach ensures the evaluation process is thorough, insightful, and reflective of the complexities inherent in diverse image segmentation tasks.

3 Crop fields detection

This section covers the history of crop field detection and the studies developed by diverse research groups to address this challenge. In sequence, we present the progress in this domain and detail how our research contributes to this context. Finally, we also present a table summarizing the most important research in this area, highlighting the important factors that supported the choices of this dissertation, such as the satellite imagery used, the machine learning models applied, and the metrics employed to assess the performance of different approaches.

3.1 Classical computer vision

[Evans et al. \(2002\)](#) used classical computer vision to introduce the canonically-guided region growing (CGRG) procedure for automated segmentation of multispectral Landsat TM images of farmland in Western Australia. This method assumes that each crop field has a single ground cover type and a known minimum width. The CGRG procedure employs a seeded region growing algorithm with internal field markers generated from a multiband, local canonical eigenvalue image. The eigenvalues discriminate between areas inside a field and at a field boundary. Comparisons with other segmentation methods indicate that CGRG generally provides more accurate results regarding field boundary position and degree of over-segmentation and under-segmentation. A sample result is demonstrated in [Figure 19](#).

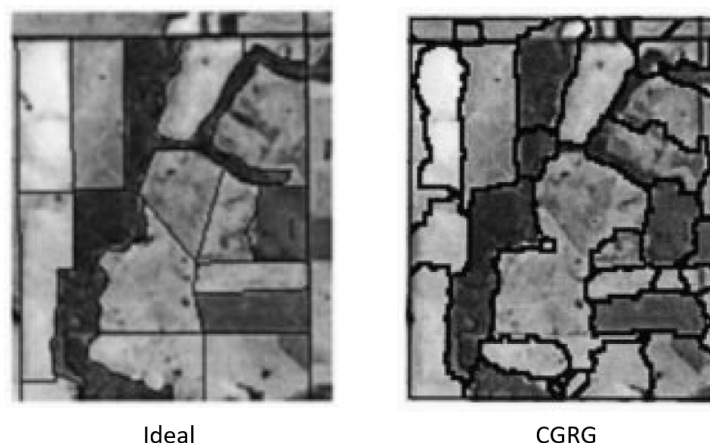


Figure 19 – Comparison between the ideal segmentation and one generated by the CGRG method. Source: [Evans et al. \(2002\)](#).

Nevertheless, it was applied to a very small and manually annotated set of images, requiring further investigation in a more broad collection of samples. A more contemporary

work led by [North, Pairman e Belliss \(2019\)](#) used classical computer vision techniques to detect the boundaries of farm fields in New Zealand. Their method identified field boundaries based on step edges or linear features in regions of low variability across the time series. The approach emphasized edge linearity over spectral differences and successfully segmented parcels with different crops and pastures, separated by features like roads and hedgerows. The method was applied to a 4,000 km² agricultural study site, demonstrating favorable results compared to existing segmentation methods regarding quantitative quality metrics and suitability for land-use classification. However, their technique still resulted in only 59% accuracy, which is insufficient for most applications. Moreover, their strategy is highly impacted by variations caused by the season in the images extracted.

3.2 Machine learning

Since classical computer vision techniques alone are often insufficient to achieve good results, further research evaluated ML models to detect crop boundaries. For instance, [Garcia-Pedrero, Gonzalo-Martin e Lillo-Saavedra \(2017\)](#) applied machine-learning algorithms to delineate agricultural parcels automatically. This method combined superpixels and supervised learning (SL) to determine which adjacent superpixels should be merged, turning the segmentation problem into a machine learning task. The diagram of the employed method is shown in Figure 20.

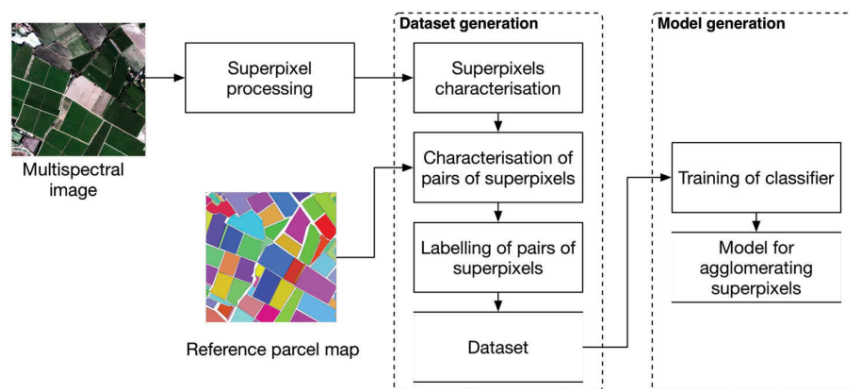


Figure 20 – Segmentation by grouping super pixels and machine learning classification. Source: [Garcia-Pedrero, Gonzalo-Martin e Lillo-Saavedra \(2017\)](#).

A visual evaluation of the methodology applied to a high-resolution satellite image of a fragmented agricultural landscape indicated good results, suggesting the potential effectiveness of machine-learning algorithms for automated parcel delineation. This work achieved an accuracy of 92% using an ensemble algorithm called RUSBoost ([SEIFFERT et al., 2010](#)) to merge superpixels and group blocks of the image, which were part of the same field. This study demonstrated how ML is a promising alternative and a precursor to further

development using DL ([GARCIA-PEDRERO; GONZALO-MARTIN; LILLO-SAAVEDRA, 2017](#)).

In the same year, [Kussul et al. \(2017\)](#) applied DL to segment and classify crop types. The architecture employed by their work included an unsupervised neural network for optical imagery segmentation and missing data restoration, along with an ensemble of supervised neural networks. The supervised networks included traditional fully connected multilayer perceptron (MLP) and random forest, which were compared with convolutional neural networks (CNNs). Experiments were conducted in a Ukrainian test site using Landsat-8 and Sentinel-1A satellite data, showing that the ensemble of CNNs outperforms MLPs, achieving accuracies exceeding 85% for major crops (wheat, maize, sunflower, soybeans, and sugar beet). The same architecture effectively discriminates summer crop types, particularly maize and soybeans. Later, [Garcia-Pedrero et al. \(2019\)](#) presented a study using open data from the Land Parcel Identification System (LPIS) in the Chartered Community of Navarre, Spain, to train a CNN model. The outlined agricultural plot boundaries generated by the CNN model were compared to those obtained using the gPb-UCM methodology used by [Crommelinck et al. \(2019\)](#), and the evaluation using the boundary displacement error index (BDE) indicates that the CNN model outperforms the gPb-UCM method. The results suggested that CNN models trained with LPIS data can be a valuable and efficient tool, reducing the need for intensive manual labor in delineating crop fields (Figure 21) ([GARCIA-PEDRERO et al., 2019](#); [GARCIA-PEDRERO; GONZALO-MARTIN; LILLO-SAAVEDRA, 2017](#); [GARCIA-PEDRERO et al., 2018](#)).

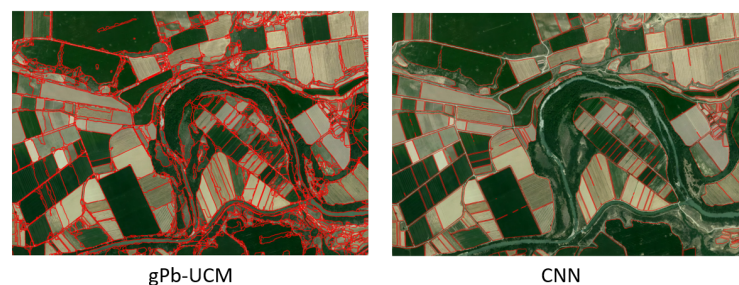


Figure 21 – Visual comparison between the segmentation results obtained with CNN and gPb-UCM. Source: [Garcia-Pedrero, Gonzalo-Martin e Lillo-Saavedra \(2017\)](#).

The latter described studies, and the good results associated with them opened a new path in this domain, where DL started to be the majority of the methods applied to address this challenge.

3.3 Deep learning

Concurrently with [Garcia-Pedrero et al. \(2019\)](#), [Persello et al. \(2019\)](#) pursued the challenge of accurately delineating agricultural fields in smallholder farms using Very

High Resolution (VHR) satellite images. These fields' irregular shape, small size, and mixed-cropping systems made automated detection challenging. The proposed strategy introduced a fully convolutional network with a globalization and grouping algorithm. The convolutional network learned complex spatial-contextual features and accurately detected sparse field contours. A hierarchical segmentation using oriented watershed transform was applied, and field segments were obtained through a combinatorial grouping algorithm. Experimental analysis in Nigeria and Mali using WorldView-2/3 images showed promising results, with the proposed strategy outperforming alternative techniques in automatically detecting and delineating field boundaries with F1-scores higher than 0.7 and 0.6 in the two test areas, respectively. In this work, however, some significant mistakes were generally associated with poor-quality training data (PERSELLO et al., 2019). The same authors extended their work to medium-resolution images from Sentinel-2. Still, they observed a severe limitation during training since the employed methods comprised many convolutional layers, resulting in a time-consuming task (PERSELLO et al., 2019; MASOUD; PERSELLO; TOLPEKIN, 2020).

Wagner e Oppelt (2020b) designed a modified version of the growing snakes active contour model based on graph theory concepts (WAGNER; OPPELT, 2020b). In sequence, Wagner e Oppelt (2020a) integrated a DL approach into their already proven graph-based model. The method achieved high results in rural areas but required further investigation of the missed boundaries, especially in areas closer to the cities (WAGNER; OPPELT, 2020a; WAGNER; OPPELT, 2020b).

Waldner e Diakogiannis (2020) formulated the task as a multi-task semantic segmentation problem using ResUNet-a, a deep convolutional neural network. The model was designed to identify the extent of fields, field boundaries, and the distance to the closest boundary simultaneously. By reconstructing three correlated outputs, the model's performance and generalization improved significantly. The approach accurately mapped field extent and boundaries using monthly composite images from Sentinel-2. The model also generalized well across different resolutions, sensors, space, and time. The convolutional neural network effectively learned complex contextual features, outperforming conventional edge filters based on classical computer vision. Moreover, they compared DL methods, extending to an improved method named Fractal-ResUNet, a network designed for semantic segmentation of agricultural images (WALDNER; DIAKOGIANNIS, 2020; WALDNER et al., 2021). The results obtained by the developed method are shown in Figure 22.

In the same year, Zhang et al. (2021) applied a very similar method, a deep semantic segmentation network named Recurrent Residual U-Net (R2U-Net), to mine low-level and deep semantic features. Additionally, a boundary connecting method was applied to integrate fragmented boundaries and generate the agricultural field boundary. The approach was tested in Heilongjiang province, China, using Sentinel-2 imagery. It outperformed

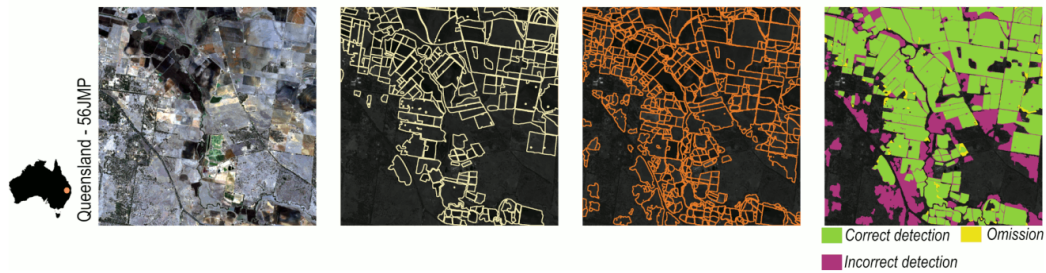


Figure 22 – Segmentation results obtained by Fractal-ResUNet method. Source: [Waldner et al. \(2021\)](#).

other methods such as U-Net, ResU-Net, traditional object-based image analysis (OBIA), and an existing global land cover map. The proposed method demonstrates improved overall accuracy (89.28%) and Kappa (0.85), suggesting great potential in agricultural field detection. Among the aforementioned techniques, R2U-Net and Fractal-ResUNet achieved the best accuracies with Sentinel-2 images. However, both depend on a large amount of training data with high-quality labeling to achieve consistent results. Moreover, they cannot distinguish a random crop field from a productive crop field, a common requirement for most agricultural applications.

One year later, to overcome the frequent need for high-quality large amounts of labeled data, other methods became the object of study, including the one led by [Li et al. \(2022a\)](#). This work recognizes the challenges of obtaining labeled samples for remote sensing image (RSI) semantic segmentation and introduces a new learning paradigm called self-supervised learning (SSL) ([CHEN et al., 2020](#)). The proposed method, Global Style and Local Matching Contrastive Learning Network (GLCNet), employs contrastive learning for semantic segmentation in RSIs. The GLCNet comprises a global style contrastive learning module to enhance image-level representation and a local feature matching contrastive learning module for pixel-level discrimination. The experimental results on four RSI semantic segmentation datasets demonstrate that GLCNet outperforms state-of-the-art SSL methods and the ImageNet pre-training method. Notably, with only 1% annotation from the original dataset, GLCNet improves Kappa by 6% on the ISPRS Potsdam dataset relative to the existing baseline. The study promoted the development of self-supervised learning in RSI semantic segmentation, highlighting its significance for tasks like global mapping. In the same year, [Wang, Waldner e Lobell \(2022\)](#) proposed a solution using transfer learning and weak supervision ([WANG et al., 2020](#)), demonstrating its success in India, where 10,000 new field labels were efficiently classified. The best model, utilizing a dataset with 1.5m resolution Airbus SPOT imagery, achieved excellent results in India. The approach involves pre-training a neural network on France field boundaries and fine-tuning on India labels. The findings suggest a scalable method for delineating crop fields in regions lacking field boundary datasets. Both results did not reach the same accuracy as previous supervised techniques; however, they addressed the limitations around

high-quality datasets. Another successful work towards the solution of limitations regarding high-quality datasets was proposed by Nascimento et al. (2023), leveraging data from machines equipped with sensors to identify whether the position captured by GNSS was a productive field. The generated dataset differed from others by focusing on productive and not generic crop fields. It was applied to modified versions of machine learning and deep learning methods derived from the state-of-the-art in their fields to demonstrate viability of the framework given important performance results with accuracies and MCC close to 97% and 0.94, respectively. Another point about this work is the employment of hexes instead of pixels to achieve the optimal balance between precision and performance. Given the promising results, this work was applied to a limited number of fields and encourages its extension to broader areas of the globe.

Finally, a recent work on the supervised learning domain applied a new technique named Multi-Swin Mask Transformer, a method based on Mask2Former (CHENG et al., 2022), an end-to-end instance segmentation framework (ZHONG et al., 2023). The method incorporates a multiscale idea into a Transformer based on Mask2Former. The study used the iFLYTEK Challenge 2021 Cultivated Land Extraction competition dataset for evaluation and compared results with Mask R-CNN, HTC, and Mask2Former, among others, that achieved relevant segmentation performances. Experimental results, demonstrated in Figure 23, indicate that MSMTransformer performs excellently, achieving high scores in segmentation accuracy.

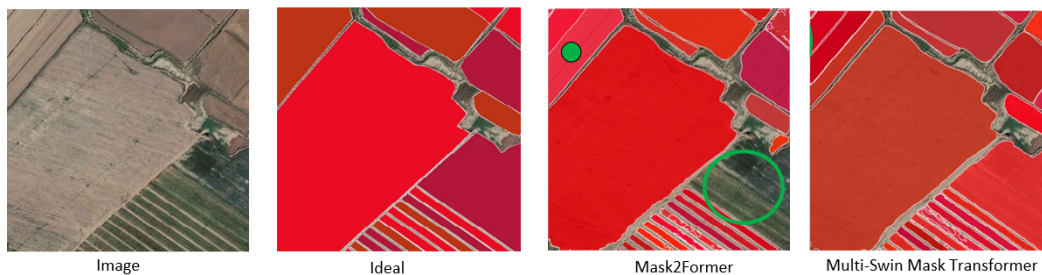


Figure 23 – Multi-Swin Mask Transformer comparison results. Source: Zhong et al. (2023).

As it has been observed, the latest studies continue to evolve both supervised and self-supervised approaches, where the dataset availability drives the trade-off. While a wide variety of datasets and metrics has been observed, the methods continue to converge to the deep learning domain, demonstrating a scenario where the results continue to be improved as the computational processing power is no longer a limitation.

3.4 Final considerations

Most existing studies are devised for detecting or segmenting all possible fields, regardless of whether they are productive. Only a few consider productive fields under

the continuous process of planting and harvesting, but even these disregard their eventual changes over time. In addition, a good-quality dataset has been observed to be a recurrent limitation. This work proposes a sustainable strategy that addresses both matters. This dissertation explores the limitations that have continuously obstructed the delivery of high-quality results by offering a solution for these limitations and employing state-of-the-art methods that match both the application and the proposed dataset. To the best of our knowledge, we are the first to propose the automatic detection of productive crop fields, taking into account the continuous agricultural cycles of planting and harvesting, employing a dataset supported by machine data and combining with timeseries of satellite images, specifically for segmentation of productive crop fields. While other studies apply timeseries to maximize the number of samples and reduce the negative effects of cloudy instances, we focus on the vegetation patterns to ensure it is a productive field. Finally, we share the results, including the dataset created to foster future research that plans to reuse the same base methodology.

3.5 Related work summary

In Table 1 we offer a different visualization of the reviewed related work, which guided this study.

Paper	Dataset	Model	Type	Year	Metrics
Nascimento et al. (2023)	Sentinel-2 combined with machine's operational data	PU Learning, Triplet Loss Siamese, Contrastive Learning	SL, SSL	2023	Accuracy, F1-Score, MCC
Zhong et al. (2023)	iFLYTEK Challenge	Multi-Swin Mask Transformer	SL	2023	Accuracy
Wang, Waldner e Lobell (2022)	Airbus SPOT SPOT-6/7 (1.5m resolution), Planet Scope (4.8m resolution)	FracTAL-ResUNet, ResUNet-a, U-Net	SL	2022	Accuracy, F1-Score and MCC
Wang et al. (2022)	CNES/Airbus Pléiades satellite (0.5m) Timeseries Sentinel 1 (10m)	U-Net, ResNet34-based U-Net, SeresNet34-based U-Net	SL	2022	Accuracy and IoU
Li et al. (2022a)	ISPRS Potsdam (5cm), DGLC (0.5m), Hubei (2m) Xiangtan	SimCLR MoCo GLCNet	SSL	2022	Accuracy and Kappa
Zhang et al. (2021)	Sentinel 2 Images (10 m, 20 m and 60 m)	ResU-Net, R2U-Net	SL	2021	Accuracy
Waldner et al. (2021)	Sentinel 2 Images (10 m, 20 m and 60 m)	FracTAL ResUNet	SL	2021	Accuracy and MCC
Taravat et al. (2021)	Sentinel 2 Images (10 m, 20 m and 60 m)	ResU-Net	SL	2021	F1-Score, Jaccard Coefficient
Fetai, Račić e Lisec (2021)	UAV images (5 cm)	U-Net ENVINet5	SL	2021	Accuracy, F1-Score, Precision and Recall

Paper	Dataset	Model	Type	Year	Metrics
Carvalho et al. (2021)	Aerial Images from government (0.24m resolution)	Panoptic-FPN	SL	2021	Accuracy and IoU
Yang et al. (2020)	WorldView-3 satellite image (0.3m resolution)	U-Net, SegNet, DenseNet	SL	2020	F1-Score, Precision and Recall
Wang et al. (2020)	Time series Landsat, Sentinel-1 and Sentinel-2 data	Pixel- and phenology-based algorithms	SL	2020	Accuracy
Waldner e Diakogiannis (2020)	Sentinel 2 Images (10 m, 20 m and 60 m)	ResUNet-a	SL	2020	Accuracy, F1-Score and MCC
Wagner e Oppelt (2020a)	Sentinel 2 Images (10 m, 20 m and 60 m)	FC MLP-NNs	SL	2020	Accuracy, F1-Score, Precision and Recall
Wagner e Oppelt (2020b)	Sentinel 2 Images (10 m, 20 m and 60 m)	Super-Pixel (CV)	SL	2020	ROI
Meyer, Lemarchand e Sidiropoulos (2020)	Sentinel 2 Images (10 m, 20 m and 60 m)	Mask R-CNN	SL	2020	Accuracy
Watkins e Niekerk (2019)	Sentinel 2 Images (10 m, 20 m and 60 m)	CV/RF	SL	2019	Overall accuracy, Kappa index, commission error and omission error.
Persello et al. (2019)	WorldView-3 satellite image (0.5m resolution)	FCN	SL	2019	Accuracy, Precision and Recall
North, Pairman e Belliss (2019)	Landsat-5,7, SPOT-4,5, Sentinel-2A and Sentinel-2B	CV	SL	2019	Accuracy
Garcia-Pedrero et al. (2019)	SIGPAC (25cm resolution)	U-Net	SL	2019	BDE Index
Ma et al. (2019)	FangChengGang (2m)	CGAN	SL	2019	Accuracy, Precision, F1-Score, Kappa Coefficient
Crommelinck et al. (2019)	Aerial image (25cm)	RF/CNN	SL	2019	Accuracy, Overlap, Precision
Xia et al. (2018)	GF-2 (0.8m resolution)	RCF/U-Net	SL	2018	IoU, Recall
Garcia-Pedrero et al. (2018)	Pléiades-1 satellite images (2m)	CV	SL	2018	BDE Index
Graesser e Ramankutty (2017)	CV	Landsat (30m)	SL	2017	Accuracy, F1-score
Garcia-Pedrero, Gonzalo-Martin e Lillo-Saavedra (2017)	WorldView-2 (4 bands) (2.4m resolution)	RUSBoost	SL	2017	Accuracy

Table 1 – Related work summary containing most important points of each study.

3.6 Challenges

Automatically detecting productive crop fields is a challenge that intersects multiple areas, and its solution could fill important gaps in agriculture management and environment control (BOLFE et al., 2020). Despite the proven benefits of fully automated crop field detection and the research already accomplished, it is still considered an open problem with significant innovation opportunities, especially when sufficient training data is unavailable (WALDNER et al., 2021; YANG et al., 2020). In almost all studies, the availability of high-quality datasets has been highlighted as one of the most hindering

limitations. Following the scarcity of datasets, no framework provides an efficient technique to combine different sources of satellite images to help addressing this limitation. Moreover, most reviewed studies focus on the segmentations of generic fields based solely on their visual shapes, not giving appropriate attention to agricultural applications that require the detection of productive fields.

4 Productive crop fields dataset

This work makes a significant contribution by developing a novel framework supported by automatically generated operational data of agricultural machines. In contrast to prior studies, the dataset employed by this framework utilizes hexagons instead of pixels, providing a well-calibrated balance between the precision necessary for this application and a reliable pixel grouping method that enhances the training algorithm’s speed, particularly for extensive regions. Moreover, adopting a unique indexing system allows the quick combination of different imagery sources with minimal modifications to the pipeline. Further details of this technique will be provided in the subsequent sections.

4.1 Dataset

The dataset is stored in tabular format, where each row corresponds to a hexagon, along with a timestamp, satellite band values, and the label.

The hexagon format was used for geo-analysis since it facilitates storing and modeling spatio-temporal data from multiple disparate sources while avoiding using rasters. It is a geospatial indexing system where, unlike others, all neighbors are equidistant. Moreover, it is beneficial to the pipeline since hexagons provide a uniform representation regardless of the position on the globe. Finally, by utilizing hexes as a common indexing system, we provide flexibility to the framework, allowing quick combination of other imagery sources to the pipeline, such as Landsat or even aerial images. This approach also eliminates the need to adjust the pipeline to different imagery precisions, enriching its architecture with scalability and maintainability.

We used the H3 Python library¹ to create such hexagons with an L12 level. At this level, each hexagon has edges of approximately 10 meters, resulting in a hexagonal area that covers 307 square meters, slightly bigger than three Sentinel-2 pixels. This resolution offers a balanced trade-off between granularity and computational efficiency while training the models. Moreover, hexagons of this size can capture fine-grained variations in satellite data within a manageable dataset size, enabling efficient processing and analysis.

In the dataset, each hexagon is associated with a timestamp. The timestamps are associated with the date and time the Sentinel-2 images were captured, providing temporal information. A time series analysis is particularly important to detect productive crop fields, given the specific patterns that distinguish a productive crop field from a non-productive one. In Figure 24, there is a sample of each of these patterns. In Figure 24a,

¹ The H3 Python library is available at <https://pypi.org/project/h3/>. Access on November 25, 2023.

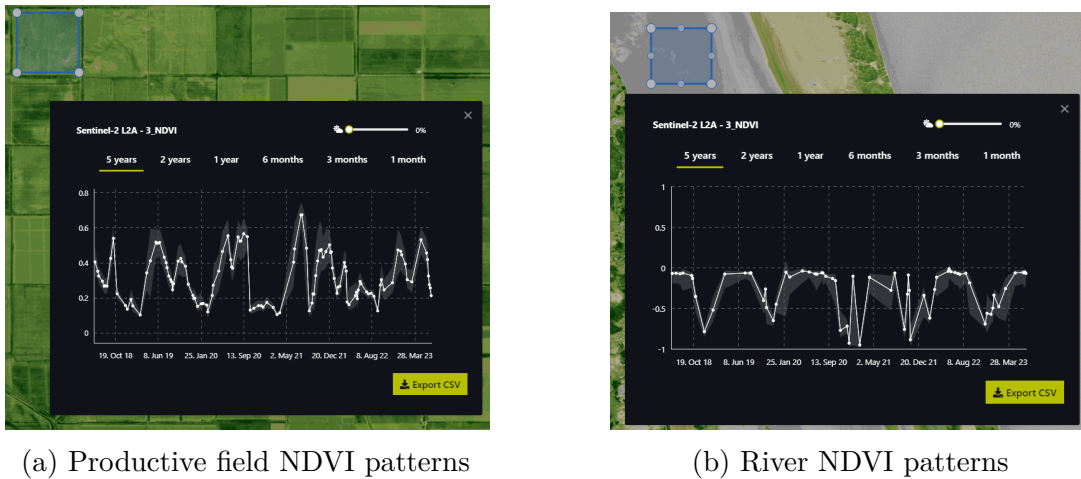


Figure 24 – NDVI patterns comparison between a productive and a non-productive area. Source: own authorship.

a productive field pattern with variations and peaks of Normalized Difference Vegetation Index (NDVI), whereas in Figure 24b, lower average levels of NDVI resulting in a different pattern that sometimes can be challenging to be distinguished by the human eye, however, can be learned by an algorithm through appropriate training. The NDVI, as a combination of specific spectral bands, although not calculated, is intrinsically contained in the values included in the dataset.

In this dataset, there are images from 2018-10-29 17:04:21 to 2019-08-15 16:59:01. The twelve band values correspond to the pixel median values in the area covered by that particular H3 hexagon. With this, we can summarize the satellite data for each hexagon in a compact and interpretable format, facilitating downstream ML analyses. Figure 25 illustrates how we create the samples used as the input for the learning models.



Figure 25 – Sequence to build a sample based upon the hexagon time series. Source: own authorship.

Lastly, we determine whether the hexagon in question is part of a productive field by inferring that if a machine executed agricultural operations, such as planting or harvesting, at a specific position (hexagon), it undoubtedly indicates a productive crop field.

This inference is made possible through the recent advancements in Precision Agriculture and the sophisticated technology embedded in these machines. Machines equipped with GNSS devices store position-related data in their internal controllers. Subsequently, this information is transmitted to a cloud server using GSM communication, commonly called the Internet of Things (IoT) (MIRAZ et al., 2015). This enables collecting geospatial data associated with machine information, finally providing the label for the hexagon in question. This process is demonstrated in the Figure 26.

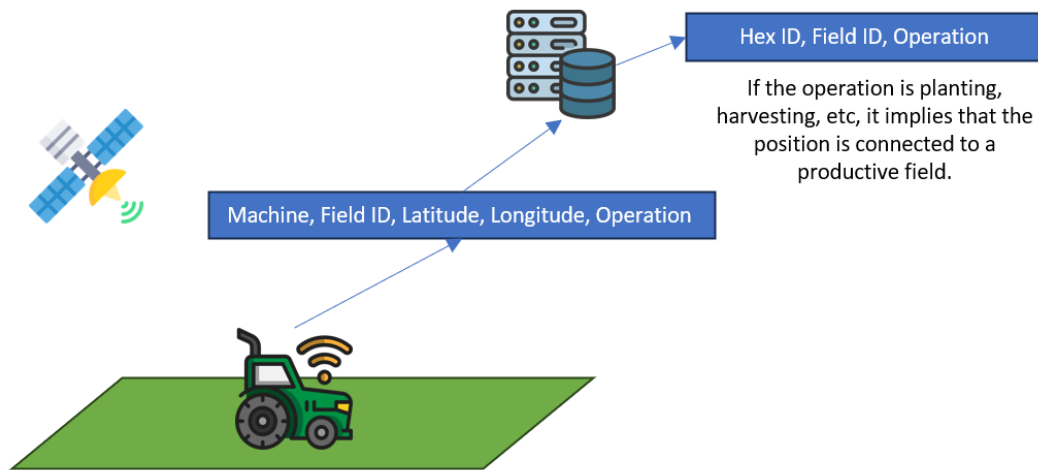


Figure 26 – Automatic labeling of hexagons based on agricultural machines operation. Source: own authorship.

The aggregation of the aforementioned information results in the provided dataset. Table 2 summarizes the dataset and its fields. The dataset includes 17 productive corn crop fields in the US within the same geographical area. These fields are geographically represented in Figure 27.

The dataset has 106,735 rows, each representing a hexagon captured on a specific date and time, followed by the 12 Sentinel-2 band values. From these, 75,990 rows are labeled as positive (i.e., productive crop fields hexagons tracked over the period) and 30,745 as negative (i.e., non-productive crop fields hexagons tracked over the period). The number of positive hexagons representing geolocations is 5,066, while the number of negative hexagons is 2,050, resulting in a slightly unbalanced dataset. The number of images for each hexagon collected on different dates varies from 15 to 38, and the averages for each field are represented in the last column.

Each hexagon has a variety of captured images associated with it. This means that, for some hexagons, we have more samples over time than others. The reason for this disparity comes from the acquisition process and satellite tiles across the globe. This difference does not interfere with the proposed method; on the contrary, it provides a larger variety of samples over time for better generalization. The distribution of the images captured at a specific time for this dataset is demonstrated in Figure 28.

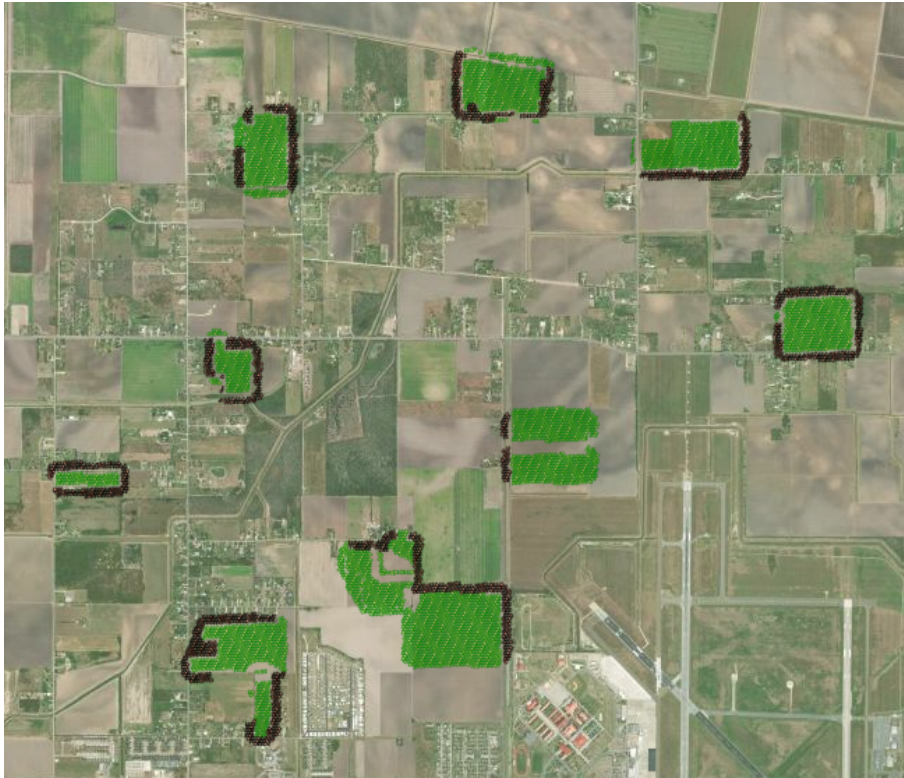


Figure 27 – Map view of the crop fields employed in this study. Source: own authorship.

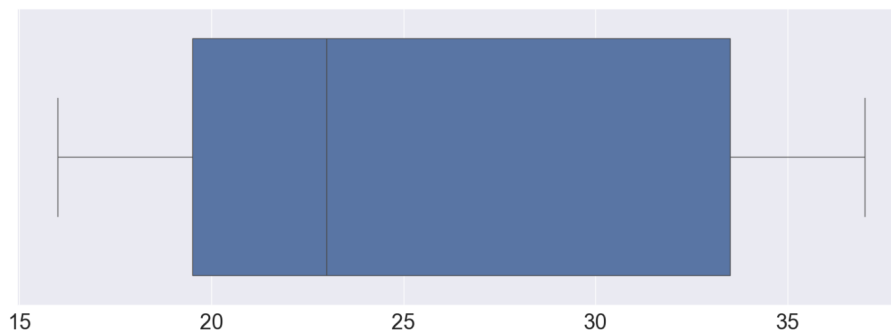


Figure 28 – Distribution of the number of samples by satellite image date. Source: own authorship.

We have labeled each hexagon as positive or negative based on the productive agriculture machine operations, such as planting or harvesting. Specifically, hexagons containing evidence of these operations were labeled positive, while neighboring hexagons within a three-layer radius were labeled as negative, as shown in Figure 29. Labeling neighboring samples as negative is particularly interesting because they are often the hardest for prediction models to classify.

One noteworthy aspect of the proposed method is its ability to distinguish non-productive fields that share the same shape as productive ones. Even a highly skilled individual would struggle to make this distinction. The key reason behind this lies in the trained model's ability to recognize patterns of band values over time from non-productive areas as distinctly different than productive cropped areas. This approach leads

Fields: 17					
Field	Hexes (samples)		Hex images collected in the period		
	positive	negative	positive	negative	average
F01	558	210	9,860	3,539	17.67 \pm 0.48
F02	734	151	14,242	2,863	19.40 \pm 0.74
F03	59	85	1,075	1,528	18.22 \pm 0.65
F04	350	150	8,326	3,518	23.79 \pm 0.48
F05	75	134	1,815	3,001	24.20 \pm 0.90
F06	288	60	4,999	1,044	17.36 \pm 0.68
F07	418	140	9,409	2,950	22.51 \pm 0.70
F08	425	154	8,584	3,084	20.20 \pm 0.82
F09	147	127	2,993	2,563	20.36 \pm 0.71
F10	369	176	6,973	3,194	18.90 \pm 0.40
F11	228	57	4,813	1,172	21.11 \pm 0.59
F12	91	96	1,701	1,738	18.69 \pm 0.53
F13	151	86	3,271	1,840	21.66 \pm 0.49
F14	275	216	4,871	3,825	17.71 \pm 0.46
F15	370	198	7,369	3,913	19.92 \pm 0.66
F16	271	25	5,030	413	18.56 \pm 0.69
F17	281	17	5,099	297	18.15 \pm 0.97

Table 2 – Productive Fields Dataset Description.



Figure 29 – Productive crop field filled with positively labeled hexagons in green surrounded by a three hexagons layer of inferred negative samples. Source: own authorship.

to significantly more accurate results. Figure 30 shows an example of a non-productive field correctly identified.

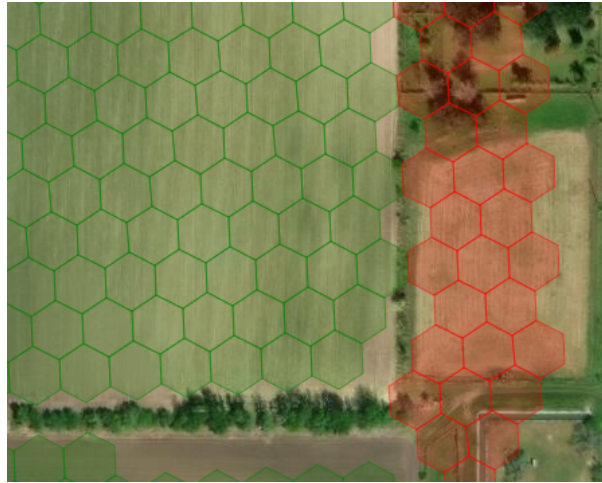


Figure 30 – A non-productive field properly classified with red hexagons. Source: own authorship.

Some false negatives may occur where productive fields are not detected due to the absence of agriculture machines equipped with satellite receptors or, for many reasons, when data are not shared. However, the dataset was manually curated to minimize the number of false negatives, specifically targeting the hexagonal regions between crop and non-crop fields. Despite these efforts, some false negatives may still be present. Even so, the proposed dataset is a valuable resource for future remote sensing research since, to our knowledge, it is the first to offer high-quality labeled productive crop fields.

4.2 Data processing

As the last step of the pipeline before training, we processed and grouped the data to construct bidimensional time series samples. Each comprises 16 randomly selected image dates, as shown in Figure 31. A total of 8,342 grouped multitemporal time series samples resulted. Selecting sparse dates throughout the year is the best strategy for a comprehensive collection comprising different weather seasons and crop stages.

We found during experimentation that the best results are achieved when selecting sparse dates along the year to have a comprehensive collection comprising different weather seasons and crop stages. We conducted two different approaches during the experiments. The first randomly distributes image dates to compose the sample, and the second distributes equally the seasons across the sample. These two approaches are demonstrated in the Figure 32.

In Figure 32a, the band values are presented without consideration for the extraction dates from Sentinel-2. In contrast, Figure 32b exhibits a structured arrangement: the first

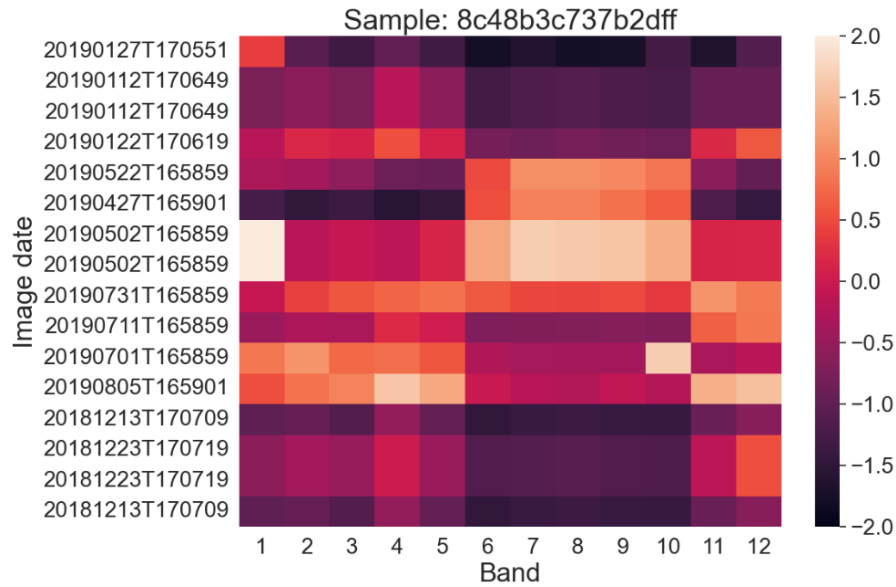


Figure 31 – Multitemporal time series sample visualization with 16 rows (i.e., each row corresponding to a unique date a satellite image was captured of the hexagon location) and 12 columns (i.e., satellite band values). Source: own authorship.

	1	2	3	4	5	6	7	8	9	10	11	12
1	DECEMBER (12)											
2	MARCH (3)											
3	FEBRUARY (2)											
4	AUGUST (8)											
5	AUGUST (8)											
6	DECEMBER (12)											
7	APRIL (4)											
8	MARCH (3)											
9	OCTOBER (10)											
10	MAY (5)											
11	MAY (5)											
12	MAY (5)											
13	APRIL (4)											
14	AUGUST (8)											
15	DECEMBER (12)											
16	DECEMBER (12)											

	1	2	3	4	5	6	7	8	9	10	11	12
1	JANUARY (1)											
2	MARCH (3)											
3	FEBRUARY (2)											
4	JANUARY (1)											
5	APRIL (4)											
6	JUNE (6)											
7	APRIL (4)											
8	MAY (5)											
9	JULY (7)											
10	AUGUST (8)											
11	AUGUST (8)											
12	JULY (7)											
13	OCTOBER (10)											
14	NOVEMBER (11)											
15	DECEMBER (12)											
16	NOVEMBER (11)											

(a) Sample with rows randomly distributed (b) Sample with rows distributed by season

Figure 32 – Methods to build samples based on months' distribution. Source: own authorship.

four rows encompass band values from the initial three months of the year, the subsequent four rows encapsulate values from the following three months, and so forth. This approach enhances the learning process of the algorithm by providing a systematically organized sample. However, a potential drawback arises when an insufficient number of months is available, an unlikely scenario given the extensive availability of Sentinel-2 images.

The original dataset was split into 80% training, 10% test, and 10% validation data, keeping the balance between positive and negative labeled data. To prevent training data leakage, we split data keeping crop fields separate into training, testing and validation. In other words, by following this practice, we avoid training the algorithm using Field 1 samples and classifying using Field 1 samples again, even if from different hexagons. The

result allows us to obtain a more robust algorithm able to classify samples from unforeseen fields.

Finally, to avoid a biased evaluation of the method's performance, we used leave-one-out cross-validation. In this context, all fields were used for training, testing and validation at least once during the experiments.

5 Benchmark results

This section opens with a description of how the selected state-of-the-art machine learning (ML) and deep learning (DL) methods, previously introduced in Chapter 1.2, were applied to accomplish our goal. Subsequently, we present the benchmark results, including the metrics that we have identified as important based on a review of the literature. All results in this section represent the averages of values obtained as a result of leave-one-out cross-validation. Both the dataset and implementation notebooks are publicly available at <https://github.com/egnascimento/productivefieldsdetection>.

5.1 Experimental protocol

This section describes the experimental protocol employed within the context of this dissertation. Based on the dataset's characteristics and training strategies, three state-of-the-art methods on ML and DL, detailed in the next sections, were selected. A brief description of the architectures and the hyperparameters utilized are also included to provide sufficient information regarding the experiments.

5.1.1 Positive Unlabeled (PU) Learning

The dataset proposed in this study comprises highly accurate positive samples and inferred negative samples. Our confidence in the accuracy of positive samples is grounded in the fact that they originated from agricultural machines reporting their current state in the assigned locations. As for negative samples, we deduced them by considering the observation that the surrounding areas of productive crop fields typically consist of non-productive zones, such as roads, forests, and rivers, unless indicated otherwise by the reporting of neighboring productive fields by the same or other machines. This particular scenario aligns effectively with PU learning.

In executing the experiments, we applied the architecture illustrated in Figure 33. For each configuration, the data underwent processing and was subsequently fed into this architecture, in a total of 17 iterations, with each iteration utilizing a different field as the test set. This application of cross-validation was crucial to prevent biased results originating from a simplistic or challenging setup. The varied testing fields ensured a robust evaluation, contributing to the reliability and generalizability of our findings.

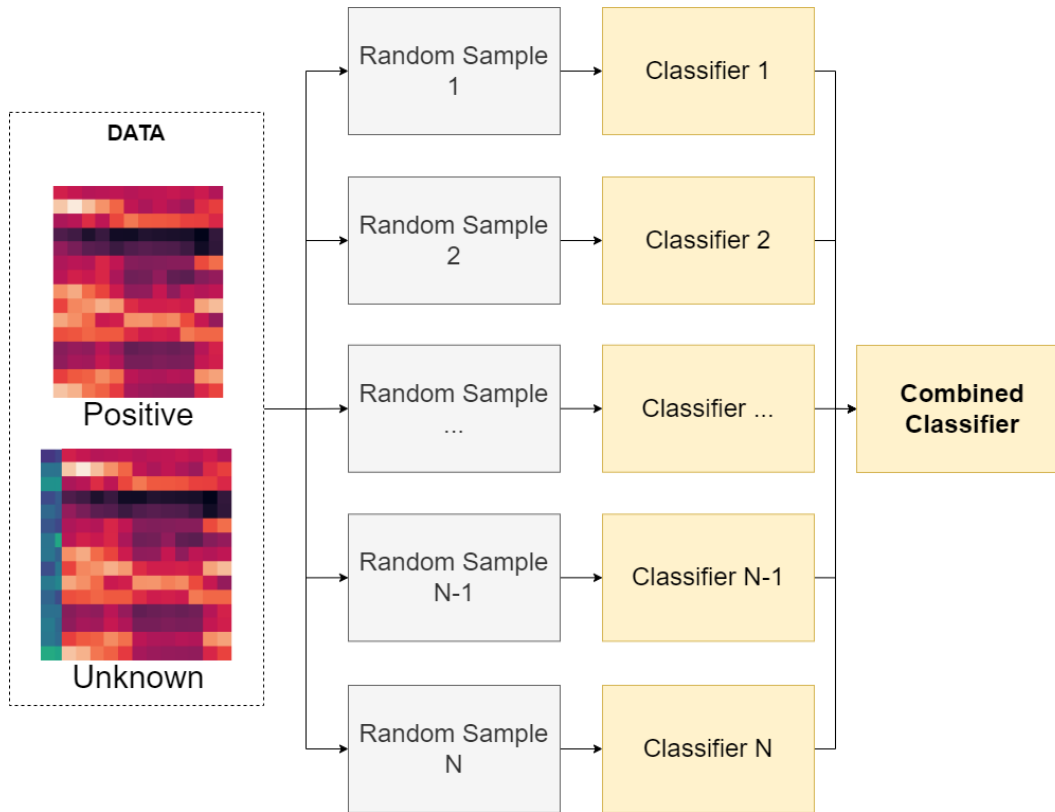


Figure 33 – PU Learning architecture employed in the benchmark testing. Source: own authorship.

5.1.2 Triplet Loss Siamese Network

Triplet Loss Siamese Networks have demonstrated success in applications related to image processing. Similarly, even though our dataset is stored in tabular format, the processed data yields a 2D matrix that resembles an image structure. This characteristic allowed us to leverage the embedding process inherent in this method.

The Triplet Loss Siamese Network architecture employed to our dataset is shown in Figure 34. In the demonstrated pipeline, a crucial component is the encoder, responsible for generating the embeddings that will subsequently be utilized in the following sections. While [Schroff, Kalenichenko e Philbin \(2015\)](#) employed complex neural network architectures for this step, we opted for a simpler approach, considering the relatively smaller complexity of our data compared to portraits. Accordingly, we implemented four convolutional layers in this step. The selection of a kernel size of 3 and a stride value of 1 was also guided by the characteristics of our processed data.

The downstream task, consisting of a fully connected neural network, was connected to the trained network to convert the results into binary classification. This facilitated the determination of whether the hexagon in question was part of a productive crop field or not.

During the experiments, it proved advantageous when dealing with datasets char-

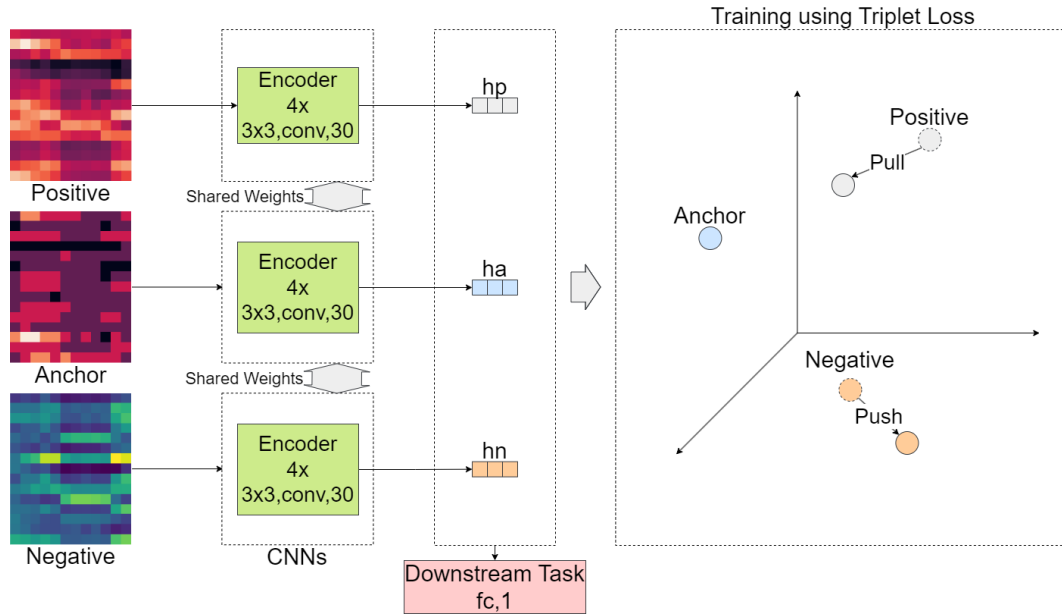


Figure 34 – Triplet Loss Siamese Networks architectures employed in the benchmark testing. Source: own authorship.

acterized by complex relationships and diverse patterns once encoded to vectors of smaller sizes, as will be further seen in the results section.

5.1.3 Contrastive Learning

As a self-supervised method, it is especially recommended in scenarios of limited data availability (e.g., regions with precision agriculture limited or nonexistent). Contrastive learning (JAISWAL et al., 2021) methods are employed to address the shortage of samples by employing data augmentation. Even though we have generated a comprehensive dataset, contrastive learning was selected to maximize the performance by augmenting our data, improving the generalization of the automatic crop field detection method.

The contrastive learning architecture employed in our experiments is demonstrated in Figure 35. Among the available implementations, we have selected the well-known SimCLR (CHEN et al., 2020) given its popularity and good performance. As Contrastive Learning is highly dependent on good data augmentation, a random jitter of up to $\pm 10\%$ was applied to all bands to create augmented samples. In this experiment, we pretrained an encoder with contrastive learning on a portion of our dataset using no labels and then fine-tuned it using only its labeled subset. As a result of this approach, we implemented a semi-supervised method that still leveraged the presence of a partial amount of labeled samples.

In contrast to the approach taken by Chen et al. (2020), who employed ResNet-50 (HE et al., 2016) for the encoder step, we opted for a more straightforward strategy. In our design, we utilized only four convolutional layers. This decision originated from

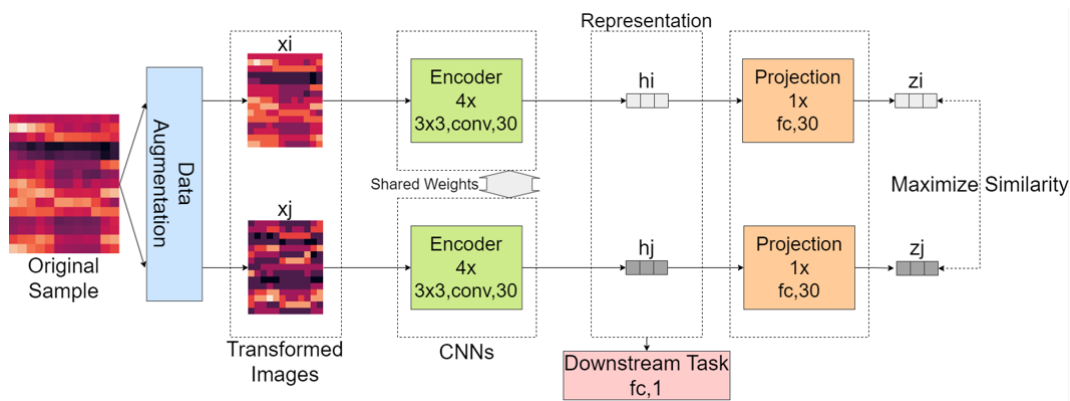


Figure 35 – Contrastive Learning architectures employed in the benchmark testing. Source: own authorship.

the relatively straightforward characteristics of our processed samples, in contrast to the complexity often present in images. In this convolutional neural network layer, the kernel size of 3 and stride of 1 were also selected based on the characteristics of our samples.

The downstream task selected, responsible for the binary classification, was a fully connected neural network. As a result, we concluded with a similar architecture to the Triplet Loss Siamese Network, allowing us to evaluate the method more than the employed configuration.

5.1.4 Metrics

To assess and compare our results with prior studies, we adopted widely accepted metrics for this application: Accuracy and F1-Score. Recognizing the inherent challenge of dealing with unbalanced datasets in this particular problem, we extended our evaluation by incorporating Matthew's correlation coefficient (MCC). Past research has noted various metrics, often influenced by the methodology and dataset specifics. Nonetheless, the consistent use of Accuracy, F1-Score, and MCC across various studies facilitates meaningful comparisons, enabling us to contextualize and benchmark our results against the broader collection of research in this field. Moreover, we applied cross-validation techniques to present metrics, ensuring unbiased results not influenced by specific test fields that may pose varying levels of difficulty or simplicity for the algorithm. To enhance method evaluation and results comprehension, we have provided a detailed representation of each leave-one-out iteration in a table, illustrated in Figure 36.

This platform incorporates a toolset, empowering researchers to identify optimal and suboptimal performance scenarios. The toolset includes features such as text filtering for efficient search, sorting by clicking on column headers, and zooming in on specific mapped areas of actual and predicted hexagons. For instance, users can examine the predominance overlay of a hexagon, offering a detailed exploration of the algorithm's performance. This platform is also openly available on the same GitHub's website where

Search



Parameters			Validation Results	Test Results				Actual	Predicted
Description	Sampling method	Bands	Accuracy	Accuracy	F1	MCC	Confusion Matrix		
Training with test on field 0 and validation on field 1 and test identifier ae26503f	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.40%	95.70%	95.77%	0.90	[199 6] [27 536]		

Figure 36 – Leave One Out iteration report. Source: own authorship.

this dissertation is stored.

5.2 Results

In this section, we present the results achieved by each benchmark method, detailing the approach and observations. Finally, we summarize the top results, including metrics suitable for evaluating this problem (ZHANG; LIU; WANG, 2018).

5.2.1 PU Learning

Firstly, when applying PU Learning, we evaluated three different underlying machine learning methods: Multi-Layer Perceptrons (MLP), Random Forests (RF) and Support Vector Machines (SVM), where the latter has been more extensively used in the literature (DENIS; GILLERON; LETOUZEY, 2005). We have also applied these combinations to two different strategies when building a sample: the first one is named “Shuffle”, where we randomly added rows to our matrix, and the second is named “Season”, where, out of the 16 rows of our matrix, we grouped each 4 rows by season. The obtained results are demonstrated in Table 3.

The best results were achieved with SVM, while better accuracies were observed in other methods due to the unbalanced nature of the dataset. This scenario highlights the importance of selecting the appropriate metrics, in this case, the MCC. In addition, the results proved that the best strategy to create samples was by Season when combined with PU Learning.

The standard deviation observed is relatively high in many cases. The reason there is an important difference between the most and least successful interactions is the algorithm’s consistency when classifying certain areas. In other words, if the method is

Model	Method	Sampling	Accuracy		F1-Score		MCC	
			%	σ	%	σ		σ
PU	SVM	Shuffle	93.17	7.4	93.14	7.54	0.83	0.15
PU	SVM	Season	95.74	2.73	95.84	2.53	0.89	0.08
PU	RF	Shuffle	92.86	6.97	92.50	7.41	0.81	0.16
PU	RF	Season	96.24	2.83	96.40	2.59	0.88	0.11
PU	MLP	Shuffle	94.12	6.65	94.18	6.71	0.84	0.16
PU	MLP	Season	93.53	7.24	94.32	5.62	0.84	0.18

Table 3 – Results achieved by the PU Learning model.

mistaken about one hexagon, it will consistently apply the same mistake to other hexagons connected to the same area, thereby the same satellite band values patterns. This effect can be observed in Figure 37.



Figure 37 – Method consistency when classifying crop fields. Source: own authorship.

While poor performance may occur due to these scenarios, it highlights the importance of adopting a cross-validation approach when assessing these methods. Additionally, errors of this nature clarify the problem’s analysis, indicating potential directions to inadequate algorithm training, for example, not including enough sample diversity. For example, insufficient inclusion of time series data into a very short period or within non-optimal intervals that do not include enough agricultural cycles can be misleading to the method to think it is a non-productive area.

The employed technique demonstrated competitive or superior performance compared to contemporary DL methods, especially in scenarios with a scarcity of positive examples and a low proportion of negatives within the unlabeled instances. Moreover, the proposed method exhibited significantly fast execution, particularly in cases where the unlabeled dataset is extensive, which can be the case depending on the satellite tile and crop field shapes.

5.2.2 Triplet Loss Siamese Networks

Secondly, we evaluated Triplet Loss Siamese Networks, converting the samples into vectors projected into the space, bringing samples of the same class together. As a deep learning method based on vector encoding, specific hyperparameters were identified to be evaluated, in this case, the vector length, computed by Cosine distance.

Other hyperparameters were also tested, but the results and overall performance were not significantly improved. The results obtained while evaluating Triplet Loss Siamese Networks are shown in Table 4.

Model	Loss	Vector	Sampling	Accuracy		F1-Score		MCC	
				%	σ	%	σ	σ	σ
TLS	Cosine	45	Shuffle	95.36	5.28	95.43	5.30	0.86	0.14
TLS	Cosine	45	Season	94.95	5.86	95.48	4.79	0.86	0.17
TLS	Cosine	30	Shuffle	95.27	5.52	95.36	5.52	0.86	0.14
TLS	Cosine	30	Season	95.56	4.16	95.90	3.49	0.87	0.14
TLS	Euclidean	30	Season	94.57	6.57	95.12	5.56	0.85	0.18
TLS	Cosine	15	Shuffle	95.15	5.18	95.21	5.19	0.86	0.14
TLS	Cosine	15	Season	94.39	6.79	94.89	6.03	0.85	0.18

Table 4 – Results achieved by Triplet Loss Siamese model.

As indicated, employing a more structured approach when building samples, the strategy previously referenced as “sampling by Season” performed better when associated with Triplet Loss Siamese Networks. A structured sample creation distributing the band values into aggregated rows keeping seasons together, consistently improved the accuracy to values close to 1%. Interestingly, regarding the encoded embedding length, also referred to as vector size, the optimal value found was 30, an intermediary number among those evaluated for this method, suggesting a length range where we contribute to the point similarity while not losing information during the encoding process. However, we observed small variations when varying the vector size. Finally, employing a different distance measurement than the one proposed by [Schroff, Kalenichenko e Philbin \(2015\)](#), in this case, the cosine distance, was beneficial to the method’s performance, resulting in accuracy improved by approximately 1%.

5.2.3 Contrastive Learning

Thirdly, a self-supervised method was evaluated, in this case, Contrastive Learning using SimCLR. It is important to highlight that, during this experiment, we did not use any labels to perform training, following the self-supervised approach. In other words, in the pipeline, we first trained the algorithm to bring similar crop field areas close together while

keeping dissimilar crop field areas apart. Finally, to evaluate the methods, we connected a simple classifier to the output of this network.

Similarly to the previous deep learning evaluation based on vector encoding, we varied the length of the vectors as well as the method of building samples. The data augmentation was performed by inserting a small jittering to the real data. We applied random jittering within the range of 30% when augmenting data for training. In contrast, we applied jittering within the range of only 10% during classification. The reasons we found to pursue this technique are:

- During training, the goal is to learn robust and generalized representations. Therefore, higher levels of data augmentation were applied to expose the model to diverse variations of the same satellite band values, making it more invariant to changes in the seasons and crop variety.
- Augmenting a higher percentage of data during training helped the model become more robust by learning to recognize essential features irrespective of variations in the input.
- If the jittering percentage applied to augmentation during training is too low, the model might become overly specialized in recognizing a particular set of samples. This can lead to overfitting and reduced generalization on new, unseen data during classification.

The optimal augmentation strategy was one of the biggest challenges when applying this method. In the literature and previous studies, it is usually applied to actual images, where augmentation can be performed by mirroring or changing the image brightness. However, we still found our approach beneficial, producing good results.

Model	Vector	Sampling	Accuracy		F1-Score		MCC	
			%	σ	%	σ	σ	
SimCLR	45	Shuffle	97.90	4.52	97.89	4.55	0.95	0.1
SimCLR	45	Season	98.69	1.74	98.69	1.78	0.96	0.06
SimCLR	30	Shuffle	97.80	4.70	97.78	4.74	0.94	0.11
SimCLR	30	Season	98.51	2.11	98.51	2.12	0.96	0.05
SimCLR	15	Shuffle	97.28	5.07	97.24	5.13	0.93	0.11
SimCLR	15	Season	98.24	2.43	98.23	2.45	0.95	0.07

Table 5 – Results achieved by SimCLR model.

In the Table 5, we show the results obtained by applying the aforementioned strategy. The best results were achieved again with sampling by Season. However, unlike

Triplet Loss Siamese Networks, the best accuracies were found using a vector size of 45. Nevertheless, the MCC was the same as a vector size of 30, suggesting a similar result.

5.2.4 Overall results

Finally, we compared the best results obtained while evaluating the proposed benchmark methods. As demonstrated in Figure 6, the accuracies of PU Learning, Triplet Loss Siamese Networks and Contrastive Learning were 95.74%, 95.56%, and 98.69%, respectively. Most importantly, when looking at the MCC values, we found in the same sequence 0.89, 0.87 and 0.96.

PU learning achieved accuracies similar to DL methods, a remarkable performance that can still be significantly improved by changing the underlying method to a DL neural network. The F1 score reinforced the observations of the accuracy results. Finally, the analysis of the Matthews correlation coefficient (MCC), appropriate to imbalanced datasets, supported the conclusion that the Contrastive Learning method achieved the best overall results, given the ability to duplicate the number of samples, improving the training, once the appropriate data augmentation method is applied.

Model	Vector	Sampling	Accuracy		F1-Score		MCC	
			%	σ	%	σ		σ
PU SVM	-	Season	95.74	2.73	95.84	2.53	0.89	0.08
TLS Cosine	30	Season	95.56	4.16	95.90	3.49	0.87	0.14
SimCLR	45	Season	98.69	1.74	98.69	1.78	0.96	0.06

Table 6 – Results achieved by the outcome prediction models.

Analyzing the results, we have observed that all models missed, in most cases, the prediction of samples located between field and non-field areas, as presented in Figure 38. These mistakes, however, should not be relevant when performing classification at the pixel level.

Additionally, the standard deviation remained relatively high across all benchmark methods evaluated, demonstrating consistency again while making mistakes predicting a hexagon and expanding these mistakes to the whole field. Nevertheless, this effect decreased with selecting the best method among those evaluated in this study. Moreover, we believe that by increasing the number of fields in the training step, we should make it more robust to these systemic misclassifications.

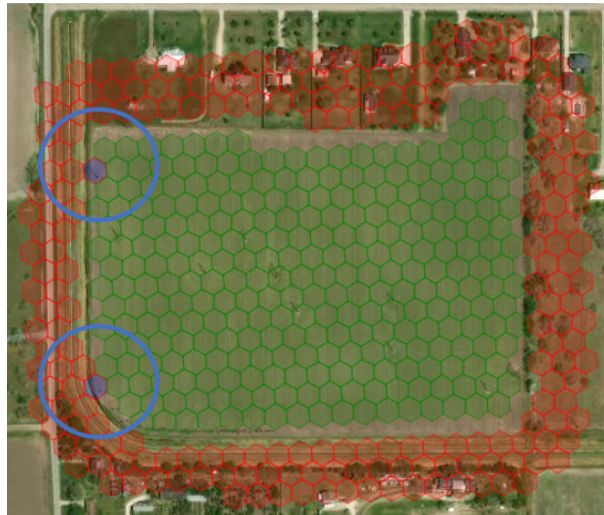


Figure 38 – Blue hexagons represent prediction mistakes, usually found between productive and non-field areas.

6 Conclusions

Precision Agriculture has experienced important advancements in recent years, marked by the continuous development of several tools and techniques. An integral aspect of precision agriculture involves the accurate segmentation of crop fields in satellite images, a task predominantly executed manually, resulting in significant expenses and time investments. Despite numerous studies to automate this intensive labor process, we acknowledged it is still a challenge to be solved. In addition, we observed the scarcity of freely available and reliable datasets aligned with this specific objective.

To contextualize our work, we introduced the fundamental aspects of machine learning and image segmentation, highlighting their interconnection and potential to provide feasible solutions in real-world scenarios. In exploring the evolution of segmentation, we traced its historical trajectory, from its origins grounded in classical computer vision techniques to its current state, where machine learning plays a crucial role. Our exploration of segmentation's historical journey highlights the shift from traditional computer vision approaches to contemporary studies, where machine learning, particularly deep learning, has emerged as a revolutionary advance in the field. Deep learning, notably, stands as the best technology in the segmentation domain, consistently delivering superior results. Today, it represents the cutting edge in image processing and segmentation, illustrating the state-of-the-art methodologies in this dynamic and rapidly advancing field.

After a thorough literature review, we identified gaps in the existing research field. We recognized that acquiring a high-quality dataset is a significant opportunity for advancement. Diverging from prior approaches vulnerable to manual errors, we strategically harnessed precision agriculture and machine-generated data to build a dataset autonomously with enhanced reliability. Having successfully curated this dataset, we applied data processing techniques to extract optimal insights. Two approaches were adopted, one randomly distributing time series images within a sample to optimize variety to the learning process and the second providing a well-structured format to the samples, aggregating the time series in a sample by season. In sequence, by adopting state-of-the-art deep learning methodologies, we aimed to harness the full potential of our dataset, ensuring its compatibility with contemporary standards and benchmarks within the field.

Lastly, we presented the results derived from applying the aforementioned methods. We have incorporated metrics widely acknowledged in prior studies to facilitate meaningful comparisons. This inclusion not only enhances the interpretability of our findings but also provides a foundation for benchmarking against existing research in the field.

The dataset proposed in this paper represents an important alternative to training

highly accurate prediction models that automatically detect productive crop fields. The main advantage is that it does not rely on manual work, which ensures, although not perfect, high-quality labeled samples in large volumes since they are continuously produced by machines equipped with precision agriculture devices. To amplify our contribution, we evaluated three different and complementary state-of-the-art ML and DL algorithms well-suited to the problem of detecting productive crop fields. The benchmark results present a performance high enough for most real applications involving detecting and delineating productive crop fields.

Future work

As our research progressed, we consistently found opportunities to refine and provide state-of-the-art solutions. While we have made significant advances, it is essential to acknowledge the prospects for future exploration that naturally emerge from the current study. Subsequently, we list these potential areas for further investigation:

- **Expand this work to broader areas.** In this study, we compiled a dataset utilizing machine-generated data, including 17 productive fields within a limited region of the United States. While this accelerated the initial phase of our research, providing quick results and offering instant insights, we acknowledge that having validated the pipeline, the logical progression is to extend the validation to a more expansive region, embracing a significantly greater number of fields. Within this broader scope, it becomes imperative to incorporate diverse crops and various geographic areas exposed to various weather conditions. Ideally, extending to other continents to maximize the diversity. This expansion is crucial to provide the algorithm with more comprehensive training, including different crops and various environmental factors, enhancing the adaptability and robustness of the algorithm.
- **Exploration of additional techniques.** In the provided context, to extend the training and help the generalization of the algorithm, additional techniques can be applied and evaluated to improve the presented benchmark results. Transfer learning (QUATTONI; COLLINS; DARRELL, 2008) and domain adaptation (LONG et al., 2015) have been successfully applied in other fields and can similarly benefit this framework.
- **Analyze this approach employing multiple satellite sources.** Our primary emphasis was on harnessing data generated by Sentinel-2 imagery. However, it is crucial to highlight that our decision to employ hexagonal grids was driven by the intrinsic ability to seamlessly integrate data from diverse image sources with multiple resolutions. Considering this, we propose extending our approach to incorporate

additional sources, such as Landsat, to enrich our dataset further. Additionally, even high-resolution imagery could be added again to provide additional diversity. This strategic integration should enhance the overall quality and richness of the dataset, allowing a more comprehensive analysis and possibly even better performances.

- **Explore additional data augmentation strategies.** Data augmentation, a mandatory component in contrastive learning methods, is typically straightforward when applied to conventional images. However, this study exemplifies that its implementation becomes more challenging when extended to alternative data sources in a tabular format. While we successfully optimized results by introducing controlled jittering into our data, we acknowledge an opportunity for exploring additional techniques that closely mimic the real-world diversity inherent in satellite imagery.
- **Experiment with additional contrastive learning implementations.** Contrastive learning emerged as the most successful method among those investigated in this study. Leveraging the widely adopted SimCLR implementation, which has demonstrated efficacy across various practical applications, contributed to our favorable outcomes. Nevertheless, we recognize the dynamic nature of contrastive learning, with new implementations continually emerging. Considering this, we recommend the replication of our established pipeline, substituting the SimCLR segment with alternative implementations such as MoCo or SwAV. This proactive approach ensures that our methodology remains at the forefront of advancements in contrastive learning, adapting to the evolving landscape of innovative techniques in the field.
- **Investigate further segmentation methods.** We introduced a segmentation methodology centered on classifying hexagonal grids within a designated region to detect productive crop fields in the same area. While this strategy simplified our study's segmentation aspect, we acknowledge opportunities for refinement. One promising opportunity for enhancement involves connecting the output of the contrastive learning network to the interconnection of hexagons. In this proposed approach, similar to graph-based segmentation, the edges connecting adjacent hexagons would represent the probability of them belonging to the same class. Potentially, this method could eliminate the need for labels, falling into the domain of fully self-supervised techniques. Exploring this approach promises important contributions for comparison with the findings presented in the current study.

Publications

The following publication is the result of this work:

- E. Nascimento, J. Just, J. Almeida and T. Almeida, “Productive Crop Field Detection: A New Dataset and Deep-Learning Benchmark Results” in **IEEE Geoscience and Remote Sensing Letters**, vol. 20, pp. 1-5, 2023, Art no. 5002005, doi: 10.1109/LGRS.2023.3296064.

Bibliography

BEUCHER, S.; MEYER, F. *The Morphological Approach to Segmentation: The Watershed Transformation*. Taylor & Francis Group, 1993. 433–481 p. ISBN 9781315214610. Disponível em: <<https://www.taylorfrancis.com/chapters/edit/10.1201/9781482277234-12/morphological-approach-segmentation-watershed-transformation-beucher-meyer>>. Cited on page 34.

BHAKTA, I.; PHADIKAR, S.; MAJUMDER, K. State-of-the-art technologies in precision agriculture: a systematic review. *Journal of the Science of Food and Agriculture*, John Wiley and Sons Ltd, v. 99, p. 4878–4888, 8 2019. ISSN 10970010. Cited on page 25.

BISHOP, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN 0387310738. Cited on page 36.

BOLFE, E. L. et al. Precision and digital agriculture: Adoption of technologies and perception of brazilian farmers. *Agriculture (Switzerland)*, MDPI AG, v. 10, p. 1–16, 12 2020. ISSN 20770472. Cited 2 times on pages 25 and 58.

BREIMAN, L. Bagging predictors. *Machine Learning*, v. 24, n. 2, p. 123–140, 1996. Cited on page 39.

CANNY, J. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8, n. 6, p. 679–698, 1986. Cited on page 35.

CARVALHO, O. L. F. de et al. Panoptic segmentation meets remote sensing. *Remote Sensing*, v. 14, p. 965, 11 2021. Cited 6 times on pages 15, 45, 46, 47, 48, and 58.

CHEN, L. C. et al. Rethinking atrous convolution for semantic image segmentation liang-chieh. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 40, 2018. ISSN 01628828. Cited 2 times on pages 44 and 46.

CHEN, T. et al. A simple framework for contrastive learning of visual representations. *37th ICML 2020*, PartF168147-3, 2020. Cited 3 times on pages 42, 55, and 71.

CHENG, B. et al. Masked-attention mask transformer for universal image segmentation. In: *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022. p. 1280–1289. Disponível em: <<https://ieeexplore.ieee.org/document/9878483>>. Cited on page 56.

CHUANG, Y.; ZHANG, S.; ZHAO, X. Deep learning-based panoptic segmentation: Recent advances and perspectives. *IET Image Processing*, v. 17, 06 2023. Cited on page 47.

CROMMELINCK, S. et al. Application of deep learning for delineation of visible cadastral boundaries from remote sensing imagery. *Remote Sensing*, MDPI AG, v. 11, p. 2505, 11 2019. ISSN 20724292. Cited 2 times on pages 53 and 58.

- DENIS, F.; GILLERON, R.; LETOUZEY, F. Learning from positive and unlabeled examples. *Theoretical Computer Science*, v. 348, n. 1, p. 70–83, 2005. ISSN 0304-3975. ALT 2000. Cited 4 times on pages 15, 27, 39, and 73.
- EVANS, C. et al. Segmenting multispectral landsat tm images into field units. *IEEE Geoscience and Remote Sensing*, v. 40, p. 1054–1064, 2002. ISSN 01962892. Cited 2 times on pages 15 and 51.
- FETAI, B.; RAČIĆ, M.; LISEC, A. Deep learning for detection of visible land boundaries from uav imagery. *Remote Sensing*, v. 13, p. 2077, 05 2021. Cited on page 57.
- GARCIA-PEDRERO, A.; GONZALO-MARTIN, C.; LILLO-SAAVEDRA, M. A machine learning approach for agricultural parcel delineation through agglomerative segmentation. *International Journal of Remote Sensing*, Taylor and Francis Ltd., v. 38, p. 1809–1819, 4 2017. ISSN 13665901. Cited 5 times on pages 15, 16, 52, 53, and 58.
- GARCIA-PEDRERO, A. et al. The outlining of agricultural plots based on spatiotemporal consensus segmentation. *Remote Sensing*, MDPI AG, v. 10, p. 1991, 12 2018. ISSN 20724292. Cited 2 times on pages 53 and 58.
- GARCIA-PEDRERO, A. et al. Deep learning for automatic outlining agricultural parcels: Exploiting the land parcel identification system. *IEEE Access*, Institute of Electrical and Electronics Engineers Inc., v. 7, p. 158223–158236, 2019. ISSN 21693536. Cited 3 times on pages 26, 53, and 58.
- GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. MIT Press, 2016. Disponível em: <<https://www.deeplearningbook.org/>>. Cited 2 times on pages 37 and 40.
- GRAESSER, J.; RAMANKUTTY, N. Detection of cropland field parcels from landsat imagery. *Remote Sensing of Environment*, Elsevier Inc., v. 201, p. 165–180, 11 2017. ISSN 00344257. Cited on page 58.
- GRIFFEL, L. M. et al. Agricultural field shape descriptors as predictors of field efficiency for perennial grass harvesting: An empirical proof. *Computers and Electronics in Agriculture*, v. 168, p. 105088, 2019. ISSN 01681699. Cited on page 25.
- GULDENRING, R.; NALPANTIDIS, L. Self-supervised contrastive learning on agricultural images. *Computers and Electronics in Agriculture*, Elsevier B.V., v. 191, p. 106510, 12 2021. ISSN 01681699. Cited on page 27.
- HAFIZ, A. M.; BHAT, G. A survey on instance segmentation: state of the art. *International Journal of Multimedia Information Retrieval*, v. 9, 09 2020. Cited on page 46.
- HE, K. et al. Mask r-cnn. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 42, n. 2, p. 386–397, 2017. Cited on page 46.
- HE, K. et al. Deep residual learning for image recognition. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2016. p. 770–778. Disponível em: <<https://ieeexplore.ieee.org/document/7780459>>. Cited on page 71.
- HERRMANN, D.; KAMPHANS, T.; LANGETEPE, E. Exploring simple triangular and hexagonal grid polygons online. 12 2010. Cited on page 32.

JAISSWAL, A. et al. A survey on contrastive self-supervised learning. *Technologies*, v. 9, 2021. Cited on page 71.

KHAN, S. S.; MADDEN, M. G. A survey of recent trends in one class classification. In: COYLE, L.; FREYNE, J. (Ed.). *Artificial Intelligence and Cognitive Science*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010. p. 188–197. ISBN 978-3-642-17080-5. Cited on page 39.

KIMERLING KEVIN SAHR, D. W. J. A.; SONG, L. Comparing geometrical properties of global grids. *Cartography and Geographic Information Science*, Taylor and Francis, v. 26, n. 4, p. 271–288, 1999. Cited on page 31.

KIRILLOV, A. et al. Panoptic segmentation. In: *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2019. p. 9396–9405. Disponível em: <<https://ieeexplore.ieee.org/document/8953237>>. Cited 2 times on pages 47 and 48.

KUMARI, P.; SINGH, Y. Delaunay triangulation coverage strategy for wireless sensor networks. In: *2012 International Conference on Computer Communication and Informatics*. IEEE, 2012. p. 1–5. Disponível em: <<https://ieeexplore.ieee.org/abstract/document/6158874>>. Cited 2 times on pages 15 and 32.

KUSSUL, N. et al. Deep learning classification of land cover and crop types using remote sensing data. *IEEE Geoscience and Remote Sensing Letters*, PP, p. 1–5, 03 2017. Cited on page 53.

LE-KHAC, P. H.; HEALY, G.; SMEATON, A. F. Contrastive representation learning: A framework and review. *IEEE Access*, v. 8, p. 193907–193934, 2020. Cited 3 times on pages 15, 42, and 43.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, Nature Publishing Group, v. 521, n. 7553, p. 436, 2015. Cited 5 times on pages 15, 37, 38, 39, and 40.

LECUN, Y. et al. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, v. 86, n. 11, p. 2278–2324, 1998. Cited on page 40.

LI, H. et al. Global and local contrastive self-supervised learning for semantic segmentation of hr remote sensing images. *IEEE Transactions on Geoscience and Remote Sensing*, v. 60, p. 1–14, 2022. Cited 2 times on pages 55 and 57.

LI, Z. et al. A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, v. 33, n. 12, p. 6999–7019, 2022. Cited 3 times on pages 15, 40, and 41.

LIN, T. et al. A survey of transformers. *AI Open*, v. 3, p. 111–132, 2022. ISSN 2666-6510. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S2666651022000146>>. Cited on page 40.

LONG, M. et al. Learning transferable features with deep adaptation networks. In: *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*. [S.l.]: JMLR.org, 2015. (ICML'15), p. 97–105. Cited on page 80.

MA, F. et al. Weakly supervised segmentation of sar imagery using superpixel and hierarchically adversarial crf. *Remote Sensing*, v. 11, n. 5, 2019. ISSN 2072-4292. Disponível em: <<https://www.mdpi.com/2072-4292/11/5/512>>. Cited on page 58.

MASOUD, K. M.; PERSELLO, C.; TOLPEKIN, V. A. Delineation of agricultural field boundaries from sentinel-2 images using a novel super-resolution contour detector based on fully convolutional networks. *Remote Sensing*, MDPI AG, v. 12, 1 2020. ISSN 20724292. Cited 2 times on pages 26 and 54.

MEYER, L.; LEMARCHAND, F.; SIDIROPOULOS, P. A deep learning architecture for batch-mode fully automated field boundary detection. In: *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences - ISPRS Archives*. International Society for Photogrammetry and Remote Sensing, 2020. v. 43, p. 1009–1016. ISSN 16821750. Disponível em: <<https://isprs-archives.copernicus.org/articles/XLIII-B3-2020/1009/2020/>>. Cited on page 58.

MINAEE, S. et al. Image segmentation using deep learning: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 44, n. 7, p. 3523–3542, 2021. Cited 2 times on pages 15 and 44.

MIRAZ, M. H. et al. A review on internet of things (iot), internet of everything (ioe) and internet of nano things (iont). In: *2015 Internet Technologies and Applications (ITA)*. IEEE, 2015. p. 219–224. Disponível em: <<https://ieeexplore.ieee.org/document/7317398>>. Cited on page 63.

MULLA, D. J. Twenty five years of remote sensing in precision agriculture: Key advances and remaining knowledge gaps. *Biosystems Engineering*, Academic Press, v. 114, p. 358–371, 2013. ISSN 15375110. Cited on page 25.

NAGENDRA, C. et al. Edge detection using fine-grained parallelism in vlsi. In: *1993 IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 1993. v. 1, p. 401–404 vol.1. Disponível em: <<https://ieeexplore.ieee.org/document/319140>>. Cited on page 35.

NAGY, B. Diagrams based on the hexagonal and triangular grids. *Acta Polytechnica Hungarica*, v. 19, 2022. ISSN 17858860. Cited 3 times on pages 15, 32, and 33.

NASCIMENTO, E. et al. Productive crop field detection: A new dataset and deep-learning benchmark results. *IEEE Geoscience and Remote Sensing Letters*, v. 20, p. 1–5, 2023. Cited 2 times on pages 56 and 57.

NELSON, G. et al. Food security, farming, and climate change to 2050: Scenarios, results, policy options. *Food Security, Farming, and Climate Change to 2050: Scenarios, Results, Policy Options*, International Food Policy Research Institute (IFPRI), 2 2010. Cited on page 25.

NIELSEN, M. A. *Neural Networks and Deep Learning*. Determination Press, 2015. Disponível em: <<http://neuralnetworksanddeeplearning.com/>>. Cited on page 37.

NORTH, H. C.; PAIRMAN, D.; BELLISS, S. E. Boundary delineation of agricultural fields in multitemporal satellite imagery. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Institute of Electrical and Electronics Engineers, v. 12, p. 237–251, 1 2019. ISSN 21511535. Cited 2 times on pages 52 and 58.

PERSELLO, C. et al. Delineation of agricultural fields in smallholder farms from satellite images using fully convolutional networks and combinatorial grouping. *Remote Sensing of Environment*, Elsevier Inc., v. 231, p. 111253, 9 2019. ISSN 00344257. Cited 4 times on pages 26, 53, 54, and 58.

QUATTONI, A.; COLLINS, M.; DARRELL, T. Transfer learning for image classification with sparse prototype representations. In: *2008 IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2008. p. 1–8. Cited on page 80.

REN, S. et al. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, v. 28, 2015. Cited on page 46.

RONNEBERGER, O.; FISCHER, P.; BROX, T. U-net: Convolutional networks for biomedical image segmentation. In: NAVAB, N. et al. (Ed.). *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015*. Springer International Publishing, 2015. p. 234–241. Disponível em: <https://link.springer.com/chapter/10.1007/978-3-319-24574-4_28>. Cited on page 45.

ROTHER, C.; KOLMOGOROV, V.; BLAKE, A. "grabcut": interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph.*, v. 23, n. 3, p. 309–314, 2004. Cited on page 34.

RYDBERG, A.; BORGEFORS, G. Integrated method for boundary delineation of agricultural fields in multispectral satellite images. *IEEE Geoscience and Remote Sensing*, v. 39, p. 2514–2520, 2001. ISSN 01962892. Cited on page 36.

SAHR, K. Hexagonal discrete global grid systems for geospatial computing. *Archives of Photogrammetry, Cartography and Remote Sensing*, v. 22, 01 2011. Cited 2 times on pages 29 and 31.

SALEHINEJAD, H. et al. Recent advances in recurrent neural networks. 12 2017. Cited on page 40.

SCHROFF, F.; KALENICHENKO, D.; PHILBIN, J. Facenet: A unified embedding for face recognition and clustering. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, IEEE, Jun 2015. Disponível em: <<http://dx.doi.org/10.1109/CVPR.2015.7298682>>. Cited 5 times on pages 15, 41, 42, 70, and 75.

SEIFFERT, C. et al. Rusboost: A hybrid approach to alleviating class imbalance. *IEEE Transactions on Systems, Man, and Cybernetics*, v. 40, p. 185–197, 2010. ISSN 10834427. Cited on page 52.

SHELHAMER, E.; LONG, J.; DARRELL, T. Fully convolutional networks for semantic segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 39, n. 4, p. 640–651, 2015. Cited on page 45.

SPEKKEN, M.; MOLIN, J. P.; ROMANELLI, T. L. Cost of boundary manoeuvres in sugarcane production. *Biosystems Engineering*, Academic Press, v. 129, p. 112–126, 1 2015. ISSN 15375110. Cited on page 25.

SUWARDI, I. S. et al. Geohash index based spatial data model for corporate. In: *2015 International Conference on Electrical Engineering and Informatics (ICEEI)*. IEEE, 2015. p. 478–483. Disponível em: <<https://ieeexplore.ieee.org/document/7352548>>. Cited on page 29.

TARAVAT, A. et al. Advanced fully convolutional networks for agricultural field boundary detection. *Remote Sensing*, MDPI AG, v. 13, p. 1–12, 2 2021. ISSN 20724292. Cited on page 57.

WAGNER, M. P.; OPPELT, N. Deep learning and adaptive graph-based growing contours for agricultural field extraction. *Remote Sensing*, MDPI AG, v. 12, p. 1990, 6 2020. ISSN 20724292. Cited 2 times on pages 54 and 58.

WAGNER, M. P.; OPPELT, N. Extracting agricultural fields from remote sensing imagery using graph-based growing contours. *Remote Sensing*, MDPI AG, v. 12, p. 1205, 4 2020. ISSN 20724292. Cited 3 times on pages 26, 54, and 58.

WALDNER, F.; DIAKOGIANNIS, F. I. Deep learning on edge: Extracting field boundaries from satellite images with a convolutional neural network. *Remote Sensing of Environment*, Elsevier Inc., v. 245, 8 2020. ISSN 00344257. Cited 3 times on pages 26, 54, and 58.

WALDNER, F. et al. Detect, consolidate, delineate: Scalable mapping of field boundaries using satellite images. *Remote Sensing*, MDPI AG, v. 13, p. 1–24, 6 2021. ISSN 20724292. Cited 6 times on pages 16, 26, 54, 55, 57, and 58.

WANG, M. et al. Agricultural field boundary delineation with satellite image segmentation for high-resolution crop mapping: A case study of rice paddy. *Agronomy*, v. 12, p. 2342, 09 2022. Cited on page 57.

WANG, S. et al. Weakly supervised deep learning for segmentation of remote sensing imagery. *Remote Sensing*, v. 12, n. 2, 2020. ISSN 2072-4292. Cited on page 55.

WANG, S.; WALDNER, F.; LOBELL, D. B. Unlocking large-scale crop field delineation in smallholder farming systems with transfer learning and weak supervision. 1 2022. Cited 2 times on pages 55 and 57.

WANG, Y. et al. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys*, v. 53, p. 1–34, 2020. ISSN 15577341. Cited on page 58.

WANG, Z.; JENSEN, J. R.; IM, J. An automatic region-based image segmentation algorithm for remote sensing applications. *Environmental Modelling and Software*, v. 25, n. 10, p. 1149–1165, 2010. ISSN 1364-8152. Cited on page 29.



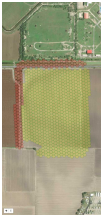
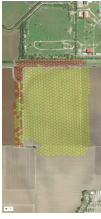










WATKINS, B.; NIEKERK, A. V. Automating field boundary delineation with multi-temporal sentinel-2 imagery. *Computers and Electronics in Agriculture*, Elsevier B.V., v. 167, p. 105078, 12 2019. ISSN 01681699. Cited on page 58.















XIA, L. et al. Deep extraction of cropland parcels from very high-resolution remotely sensed imagery. In: *2018 7th International Conference on Agro-geoinformatics*. IEEE, 2018. p. 1–5. Disponível em: <<https://ieeexplore.ieee.org/document/8476002>>. Cited on page 58.



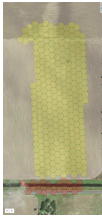
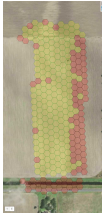
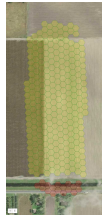
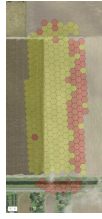
- XIAO, P. et al. Cosegmentation for object-based building change detection from high-resolution remotely sensed images. *IEEE Transactions on Geoscience and Remote Sensing*, PP, p. 1–17, 01 2017. Cited 2 times on pages 15 and 34.
- YANG, R. et al. Detecting functional field units from satellite images in smallholder farming systems using a deep learning based computer vision approach: A case study from bangladesh. *Remote Sensing Applications: Society and Environment*, v. 20, p. 100413, 2020. ISSN 2352-9385. Cited on page 58.
- YUE, M. et al. A precision evaluation index system for remote sensing data sampling based on hexagonal discrete grids. *ISPRS International Journal of Geo-Information*, v. 10, p. 194, 03 2021. Cited 2 times on pages 32 and 33.
- ZHANG, H. et al. Automated delineation of agricultural field boundaries from sentinel-2 images using recurrent residual u-net. *International Journal of Applied Earth Observation and Geoinformation*, Elsevier B.V., v. 105, p. 102557, 12 2021. ISSN 1872826X. Cited 2 times on pages 54 and 57.
- ZHANG, N.; WANG, M.; WANG, N. Precision agriculture - a worldwide overview. *Computers and Electronics in Agriculture*, Elsevier, v. 36, p. 113–132, 11 2002. ISSN 01681699. Cited on page 25.
- ZHANG, Z.; LIU, Q.; WANG, Y. Road extraction by deep residual u-net. *IEEE Geoscience and Remote Sensing Letters*, v. 15, n. 5, p. 749–753, 2018. Cited on page 73.
- ZHENG, T. et al. Research on distance transform and neural network lidar information sampling classification-based semantic segmentation of 2d indoor room maps. *Sensors*, v. 21, p. 1365, 02 2021. Cited 2 times on pages 15 and 34.
- ZHONG, B. et al. Multi-swin mask transformer for instance segmentation of agricultural field extraction. *Remote Sensing*, v. 15, p. 549, 01 2023. Cited 3 times on pages 16, 56, and 57.



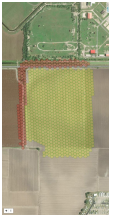
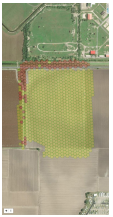










APPENDIX A – Cross validation iterations















In this chapter, we present a more detailed representation of each benchmark method iteration, integral to the previously discussed results. The tables presented herein offer a granular view of each method and their various combinations, playing a crucial role in refining strategies and hyperparameters. These reports are also accessible on the website <https://egnascimento.github.io/fieldboundarydetection/report/index.html>, enabling researchers to easily sort and filter results. Moreover, the platform allows for a closer examination of images, facilitating an in-depth understanding of both successful and unsuccessful instances in each iteration.




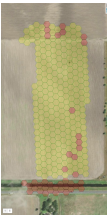
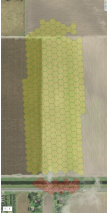
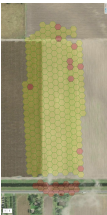
PU Learning with MLP - Parameters				Test Results					
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier ae26503f	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.40%	95.70%	95.77%	0.9	$\begin{bmatrix} 199 & 6 \\ 27 & 536 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier f878b227	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	79.81%	95.71%	95.47%	0.84	$\begin{bmatrix} 111 & 36 \\ 2 & 736 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 912f44bf	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.39%	88.46%	88.47%	0.77	$\begin{bmatrix} 44 & 5 \\ 7 & 48 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 2e074526	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.78%	94.99%	95.10%	0.89	$\begin{bmatrix} 138 & 1 \\ 24 & 336 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 9387c3f5	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	83.53%	97.61%	97.60%	0.95	$\begin{bmatrix} 133 & 1 \\ 4 & 71 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier c07968f8	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.39%	81.44%	84.20%	0.54	$\begin{bmatrix} 40 & 2 \\ 60 & 232 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 4af3719a	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.06%	98.92%	98.92%	0.97	$\begin{bmatrix} 133 & 6 \\ 0 & 419 \end{bmatrix}$		



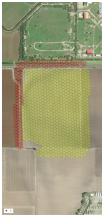
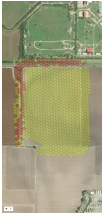










PU Learning with MLP - Parameters				Test Results					
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 352d8ab6	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.08%	98.45%	98.44%	0.96	$\begin{bmatrix} 147 & 6 \\ 3 & 423 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 9309ecb0	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.12%	96.35%	96.35%	0.93	$\begin{bmatrix} 124 & 3 \\ 7 & 140 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 81738b7e	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.48%	98.12%	98.12%	0.96	$\begin{bmatrix} 166 & 3 \\ 7 & 355 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 710c080e	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.68%	97.54%	97.49%	0.92	$\begin{bmatrix} 50 & 6 \\ 1 & 227 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier f123ad56	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	96.22%	96.22%	0.92	$\begin{bmatrix} 93 & 3 \\ 4 & 85 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 2863c2b3	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.96%	98.73%	98.73%	0.97	$\begin{bmatrix} 84 & 2 \\ 1 & 150 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier ae2a167f	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.30%	96.95%	96.93%	0.94	$\begin{bmatrix} 201 & 15 \\ 0 & 275 \end{bmatrix}$		












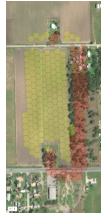


PU Learning with MLP - Parameters				Validation				Test Results			
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted		
Training with test on field 14 and validation on field 15 and test identifier a77eb15e	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	82.09%	99.65%	99.65%	0.99	$\begin{bmatrix} 198 & 0 \\ 2 & 368 \end{bmatrix}$				
Training with test on field 15 and validation on field 16 and test identifier 5ace0b2f	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	76.51%	76.69%	81.75%	0.45	$\begin{bmatrix} 25 & 0 \\ 69 & 202 \end{bmatrix}$				
Training with test on field 16 and validation on field 0 and test identifier d0cf9d06	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.35%	78.52%	84.16%	0.4	$\begin{bmatrix} 17 & 0 \\ 64 & 217 \end{bmatrix}$				



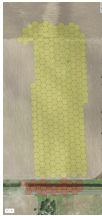
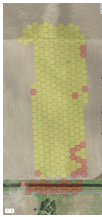
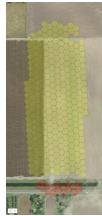
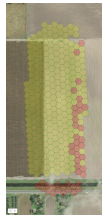
PU Learning with MLP - Parameters			Test Results						
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 5c878cfd	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.54%	98.04%	98.05%	0.95	$\begin{bmatrix} 200 & 4 \\ 11 & 552 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 64b1594e	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	68.91%	94.54%	94.08%	0.79	$\begin{bmatrix} 96 & 48 \\ 0 & 735 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier c0d81fa0	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.79%	71.43%	70.94%	0.44	$\begin{bmatrix} 34 & 24 \\ 10 & 51 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier e63adab3	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	89.00%	94.19%	94.32%	0.87	$\begin{bmatrix} 137 & 2 \\ 27 & 333 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier c902519a	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	86.46%	88.04%	87.43%	0.75	$\begin{bmatrix} 133 & 1 \\ 24 & 51 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 8fdbbc063	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.28%	86.46%	87.86%	0.49	$\begin{bmatrix} 26 & 10 \\ 34 & 255 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 2fd054fa	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.58%	98.75%	98.73%	0.97	$\begin{bmatrix} 132 & 7 \\ 0 & 419 \end{bmatrix}$		



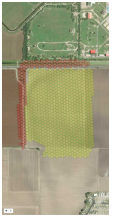








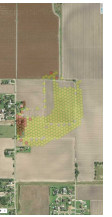


PU Learning with MLP - Parameters			Test Results					
Description	Sampling	Bands	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier e2510890	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.81%	96.86%	0.92	$\begin{bmatrix} 140 & 13 \\ 5 & 421 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 778825cb	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	91.84%	96.71%	0.93	$\begin{bmatrix} 121 & 6 \\ 3 & 144 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier bb5c11cc	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.59%	93.46%	0.86	$\begin{bmatrix} 174 & 0 \\ 36 & 329 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 7bb34f17	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.48%	98.23%	0.94	$\begin{bmatrix} 53 & 3 \\ 2 & 226 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 6869b122	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.78%	97.17%	0.94	$\begin{bmatrix} 90 & 1 \\ 4 & 82 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier f62ca564	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.76%	97.49%	0.95	$\begin{bmatrix} 86 & 0 \\ 6 & 145 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 68a6dcd9	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.47%	98.17%	0.96	$\begin{bmatrix} 211 & 5 \\ 4 & 271 \end{bmatrix}$		















PU Learning with MLP - Parameters				Test Results					
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 5677d8a	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.92%	99.47%	99.47%	0.99	$\begin{bmatrix} 196 & 2 \\ 1 & 369 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier f09f3371	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.63%	94.58%	95.17%	0.75	$\begin{bmatrix} 24 & 0 \\ 16 & 255 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 0423237f	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.39%	96.63%	96.99%	0.78	$\begin{bmatrix} 17 & 0 \\ 10 & 270 \end{bmatrix}$		





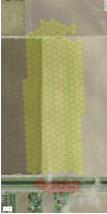
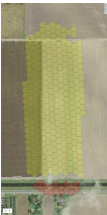
PU Learning with Random Forest - Parameters				Validation				Test Results			
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted		
Training with test on field 0 and validation on field 1 and test identifier faee9ef3	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.03%	93.36%	93.19%	0.83	$\begin{bmatrix} 165 & 40 \\ 11 & 552 \end{bmatrix}$				
Training with test on field 1 and validation on field 2 and test identifier 221a77c5	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	85.58%	94.58%	94.34%	0.79	$\begin{bmatrix} 109 & 38 \\ 10 & 728 \end{bmatrix}$				
Training with test on field 2 and validation on field 3 and test identifier 6240ec5d	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.60%	91.35%	91.35%	0.83	$\begin{bmatrix} 46 & 3 \\ 6 & 49 \end{bmatrix}$				
Training with test on field 3 and validation on field 4 and test identifier 1ae290fb	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.13%	98.00%	97.98%	0.95	$\begin{bmatrix} 131 & 8 \\ 2 & 358 \end{bmatrix}$				
Training with test on field 4 and validation on field 5 and test identifier 85eb09bc	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.22%	97.61%	97.60%	0.95	$\begin{bmatrix} 132 & 2 \\ 3 & 72 \end{bmatrix}$				
Training with test on field 5 and validation on field 6 and test identifier 2d7932ca	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.85%	91.32%	92.07%	0.7	$\begin{bmatrix} 39 & 3 \\ 26 & 266 \end{bmatrix}$				
Training with test on field 6 and validation on field 7 and test identifier d48ada61	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.75%	98.57%	98.56%	0.96	$\begin{bmatrix} 132 & 7 \\ 1 & 418 \end{bmatrix}$				











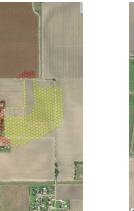
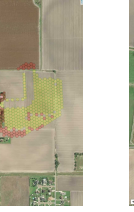


PU Learning with Random Forest - Parameters				Validation				Test Results			
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted		
Training with test on field 7 and validation on field 8 and test identifier 98333cd2	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.81%	97.81%	96.87%	0.92	$\begin{bmatrix} 141 & 12 \\ 6 & 420 \end{bmatrix}$				
Training with test on field 8 and validation on field 9 and test identifier f3c550a6	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	97.81%	97.81%	0.96	$\begin{bmatrix} 123 & 4 \\ 2 & 145 \end{bmatrix}$				
Training with test on field 9 and validation on field 10 and test identifier 499e0727	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.59%	98.31%	98.32%	0.96	$\begin{bmatrix} 169 & 0 \\ 9 & 353 \end{bmatrix}$				
Training with test on field 10 and validation on field 11 and test identifier f013007d	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.84%	98.24%	98.22%	0.94	$\begin{bmatrix} 52 & 4 \\ 1 & 227 \end{bmatrix}$				
Training with test on field 11 and validation on field 12 and test identifier 8e589ce9	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	97.84%	97.84%	0.96	$\begin{bmatrix} 95 & 1 \\ 3 & 86 \end{bmatrix}$				
Training with test on field 12 and validation on field 13 and test identifier cac5c8cb	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.17%	98.73%	98.73%	0.97	$\begin{bmatrix} 83 & 3 \\ 0 & 151 \end{bmatrix}$				
Training with test on field 13 and validation on field 14 and test identifier 7acbd57	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.65%	98.37%	98.37%	0.97	$\begin{bmatrix} 208 & 8 \\ 0 & 275 \end{bmatrix}$				


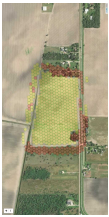












PU Learning with Random Forest - Parameters				Test Results					
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 4e358ecd	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.58%	99.65%	99.65%	0.99	$\begin{bmatrix} 197 & 1 \\ 1 & 369 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 70a1f684	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	86.58%	94.59%	95.17%	0.76	$\begin{bmatrix} 25 & 0 \\ 16 & 255 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 835f229c	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.05%	90.94%	92.72%	0.59	$\begin{bmatrix} 17 & 0 \\ 27 & 254 \end{bmatrix}$		





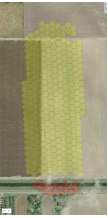

PU Learning with Random Forest - Parameters			Test Results						
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier f6adb499	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	87.37%	88.92%	88.17%	0.7	$\begin{bmatrix} 129 & 75 \\ 10 & 553 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 11cf0cbe	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	68.91%	88.51%	86.27%	0.51	$\begin{bmatrix} 49 & 95 \\ 6 & 729 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 8034ff9c	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.79%	69.75%	68.04%	0.43	$\begin{bmatrix} 27 & 31 \\ 5 & 56 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 335572db	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	83.25%	94.19%	93.98%	0.86	$\begin{bmatrix} 111 & 28 \\ 1 & 359 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier e05082a7	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.77%	85.17%	85.45%	0.71	$\begin{bmatrix} 108 & 26 \\ 5 & 70 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 7101cd85	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.11%	94.77%	94.60%	0.72	$\begin{bmatrix} 25 & 11 \\ 6 & 283 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 5a401728	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.96%	90.86%	90.10%	0.75	$\begin{bmatrix} 88 & 51 \\ 0 & 419 \end{bmatrix}$		



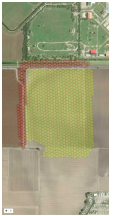
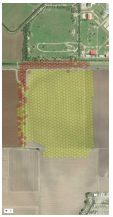







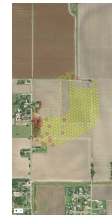


PU Learning with Random Forest - Parameters			Test Results						
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 252192dc	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.70%	94.13%	93.87%	0.85	$\begin{bmatrix} 119 & 34 \\ 0 & 426 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 18f843e2	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.36%	93.43%	93.38%	0.87	$\begin{bmatrix} 110 & 17 \\ 1 & 146 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier c3f585ad	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.42%	95.92%	95.97%	0.91	$\begin{bmatrix} 172 & 2 \\ 20 & 345 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 8f5a8dff	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.18%	94.01%	93.65%	0.8	$\begin{bmatrix} 40 & 16 \\ 1 & 227 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier e9b6ec85	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.62%	96.61%	96.61%	0.93	$\begin{bmatrix} 87 & 4 \\ 2 & 84 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 7fb1ea7b	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.96%	96.62%	96.65%	0.93	$\begin{bmatrix} 85 & 1 \\ 7 & 144 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 1ecb91ee	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.77%	98.78%	98.78%	0.98	$\begin{bmatrix} 213 & 3 \\ 3 & 272 \end{bmatrix}$		















PU Learning with Random Forest - Parameters			Test Results						
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 3c4cb99f	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.64%	98.24%	98.23%	0.96	$\begin{bmatrix} 188 & 10 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier ad5caaf8	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.64%	98.67%	0.91	$\begin{bmatrix} 23 & 1 \\ 3 & 268 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 9fb04698	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	89.18%	100.00%	100.00%	1.0	$\begin{bmatrix} 17 & 0 \\ 0 & 280 \end{bmatrix}$		




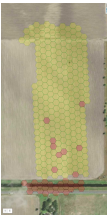
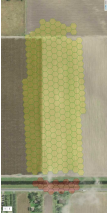
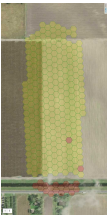
PU Learning with SVC - Parameters			Validation					Test Results				
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted			
Training with test on field 0 and validation on field 1 and test identifier 478125f3	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.53%	94.27%	94.28%	0.86	$\begin{bmatrix} 189 & 21 \\ 23 & 535 \end{bmatrix}$					
Training with test on field 1 and validation on field 2 and test identifier dc381b4f	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	84.07%	98.87%	98.86%	0.96	$\begin{bmatrix} 143 & 8 \\ 2 & 732 \end{bmatrix}$					
Training with test on field 2 and validation on field 3 and test identifier ce9d636f	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.00%	86.73%	86.78%	0.73	$\begin{bmatrix} 58 & 9 \\ 6 & 40 \end{bmatrix}$					
Training with test on field 3 and validation on field 4 and test identifier e88f4bf2	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.78%	97.00%	97.02%	0.93	$\begin{bmatrix} 147 & 3 \\ 12 & 338 \end{bmatrix}$					
Training with test on field 4 and validation on field 5 and test identifier 0aae9cf9	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	90.77%	93.78%	93.73%	0.86	$\begin{bmatrix} 130 & 4 \\ 9 & 66 \end{bmatrix}$					
Training with test on field 5 and validation on field 6 and test identifier b6009f1c	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.21%	91.37%	91.96%	0.74	$\begin{bmatrix} 50 & 3 \\ 26 & 257 \end{bmatrix}$					
Training with test on field 6 and validation on field 7 and test identifier 11848128	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.89%	97.85%	97.82%	0.94	$\begin{bmatrix} 128 & 12 \\ 0 & 418 \end{bmatrix}$					



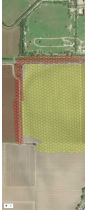
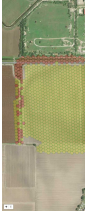








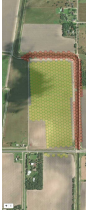
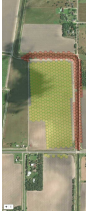
PU Learning with SVC - Parameters			Validation					Test Results			
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted		
Training with test on field 7 and validation on field 8 and test identifier 07c24f68	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.81%	97.58%	97.59%	0.94	$\begin{bmatrix} 148 & 6 \\ 8 & 417 \end{bmatrix}$				
Training with test on field 8 and validation on field 9 and test identifier d0d9f6a8	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.99%	97.45%	97.45%	0.95	$\begin{bmatrix} 126 & 1 \\ 6 & 141 \end{bmatrix}$				
Training with test on field 9 and validation on field 10 and test identifier 8c9c3850	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.19%	97.93%	97.94%	0.95	$\begin{bmatrix} 166 & 3 \\ 8 & 355 \end{bmatrix}$				
Training with test on field 10 and validation on field 11 and test identifier 4083ecc8	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.33%	96.84%	96.76%	0.9	$\begin{bmatrix} 49 & 8 \\ 1 & 227 \end{bmatrix}$				
Training with test on field 11 and validation on field 12 and test identifier efd336e5	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.73%	97.33%	97.32%	0.95	$\begin{bmatrix} 95 & 1 \\ 4 & 87 \end{bmatrix}$				
Training with test on field 12 and validation on field 13 and test identifier f736cf77	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.96%	99.58%	99.58%	0.99	$\begin{bmatrix} 85 & 1 \\ 0 & 151 \end{bmatrix}$				
Training with test on field 13 and validation on field 14 and test identifier c19b2747	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.65%	98.57%	98.57%	0.97	$\begin{bmatrix} 210 & 6 \\ 1 & 274 \end{bmatrix}$				















PU Learning with SVC - Parameters				Test Results					
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 5a166de2	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	86.49%	99.65%	99.65%	0.99	$\begin{bmatrix} 197 & 1 \\ 1 & 369 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier f0318221	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	78.52%	80.07%	84.24%	0.48	$\begin{bmatrix} 25 & 0 \\ 59 & 212 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier a68d193b	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.44%	82.89%	87.17%	0.45	$\begin{bmatrix} 17 & 0 \\ 51 & 230 \end{bmatrix}$		





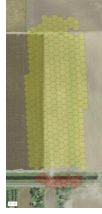
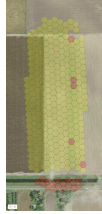
PU Learning with SVC - Parameters			Test Results						
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 785052cb	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.88%	95.57%	95.55%	0.89	$\begin{bmatrix} 185 & 19 \\ 15 & 548 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier d624d6cf	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	68.07%	95.45%	95.21%	0.83	$\begin{bmatrix} 108 & 36 \\ 4 & 731 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 3e7e7a55	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.19%	68.07%	67.36%	0.37	$\begin{bmatrix} 31 & 27 \\ 11 & 50 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 000d138c	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	84.69%	96.59%	96.60%	0.92	$\begin{bmatrix} 131 & 8 \\ 9 & 351 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier bde9224a6	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	91.69%	83.73%	83.51%	0.64	$\begin{bmatrix} 121 & 13 \\ 21 & 54 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 358876d7	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.59%	92.62%	92.86%	0.65	$\begin{bmatrix} 27 & 9 \\ 15 & 274 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier cb9fd9f	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.61%	96.06%	95.95%	0.89	$\begin{bmatrix} 118 & 21 \\ 1 & 418 \end{bmatrix}$		



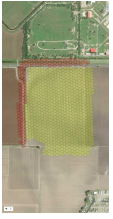


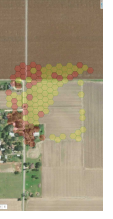




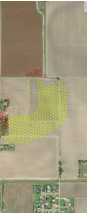
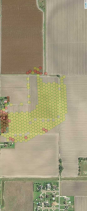


PU Learning with SVC - Parameters			Test Results						
Description	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 2ce6f577	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.18%	93.96%	93.72%	0.84	$\begin{bmatrix} 120 & 33 \\ 2 & 424 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 4bc09320	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	86.46%	98.18%	98.17%	0.96	$\begin{bmatrix} 122 & 5 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 16534887	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.83%	87.57%	87.92%	0.76	$\begin{bmatrix} 170 & 4 \\ 63 & 302 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier ff8ed500	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.35%	97.54%	97.53%	0.92	$\begin{bmatrix} 52 & 4 \\ 3 & 225 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 8b337829	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.25%	94.92%	94.92%	0.9	$\begin{bmatrix} 86 & 5 \\ 4 & 82 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier b4cd4b15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.52%	91.98%	92.10%	0.84	$\begin{bmatrix} 85 & 1 \\ 18 & 133 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 04e01d13	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.12%	95.93%	95.94%	0.92	$\begin{bmatrix} 215 & 1 \\ 19 & 256 \end{bmatrix}$		















PU Learning with SVC - Parameters				Test Results					
Description	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier b8e2203e	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.63%	99.47%	99.47%	0.99	$\begin{bmatrix} 195 & 3 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 90e1a2ed	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.33%	96.95%	97.16%	0.84	$\begin{bmatrix} 24 & 0 \\ 9 & 262 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier b0bd0529	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.70%	99.33%	99.34%	0.94	$\begin{bmatrix} 17 & 0 \\ 2 & 278 \end{bmatrix}$		







Contrastive Learning with Temperature 0.1 - Parameters				Validation				Test Results			
Description	Width	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted	
Training with test on field 0 and validation on field 1 and test identifier e77b3feb	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.06%	95.96%	95.94%	0.9	$\begin{bmatrix} 186 & 19 \\ 12 & 551 \end{bmatrix}$			
Training with test on field 1 and validation on field 2 and test identifier 90e5786b	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.31%	95.03%	94.72%	0.81	$\begin{bmatrix} 106 & 41 \\ 3 & 735 \end{bmatrix}$			
Training with test on field 2 and validation on field 3 and test identifier 0d5be044	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.80%	90.38%	90.39%	0.81	$\begin{bmatrix} 46 & 3 \\ 7 & 48 \end{bmatrix}$			
Training with test on field 3 and validation on field 4 and test identifier 79f820d2	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.80%	98.79%	0.97	$\begin{bmatrix} 135 & 4 \\ 2 & 358 \end{bmatrix}$			
Training with test on field 4 and validation on field 5 and test identifier c44cbfc9	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.70%	100.00%	100.00%	1.0	$\begin{bmatrix} 134 & 0 \\ 0 & 75 \end{bmatrix}$			
Training with test on field 5 and validation on field 6 and test identifier 1bb97acc	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	96.71%	96.78%	0.86	$\begin{bmatrix} 39 & 3 \\ 8 & 284 \end{bmatrix}$			
Training with test on field 6 and validation on field 7 and test identifier 28e998c3	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 139 & 0 \\ 0 & 419 \end{bmatrix}$			



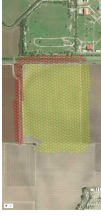
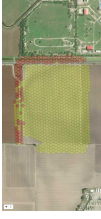










Contrastive Learning with Temperature 0.1 - Parameters					Test Results					
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 86d7a9f6	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.31%	99.31%	0.98	$\begin{bmatrix} 151 & 2 \\ 2 & 424 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier dac054ce	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 127 & 0 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier eace0007	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.44%	99.44%	0.99	$\begin{bmatrix} 169 & 0 \\ 3 & 359 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier c823f114	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.92%	99.30%	99.30%	0.98	$\begin{bmatrix} 55 & 1 \\ 1 & 227 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 074ec227	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.92%	98.92%	0.98	$\begin{bmatrix} 95 & 1 \\ 1 & 88 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 8003d9b4	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.58%	99.58%	0.99	$\begin{bmatrix} 85 & 1 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier fd37ef72	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.98%	98.98%	0.98	$\begin{bmatrix} 211 & 5 \\ 0 & 275 \end{bmatrix}$		















Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 58767858	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 3ba11e26	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.32%	99.34%	0.96	$\begin{bmatrix} 25 & 0 \\ 2 & 269 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier af3f920f	15	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.74%	98.32%	98.42%	0.87	$\begin{bmatrix} 17 & 0 \\ 5 & 276 \end{bmatrix}$		





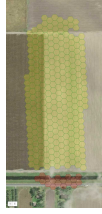
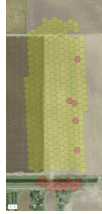
Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 881bbedc	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.02%	97.78%	97.79%	0.94	$\begin{bmatrix} 198 & 6 \\ 11 & 552 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier a6e1b0b8	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	84.87%	93.74%	93.12%	0.76	$\begin{bmatrix} 89 & 55 \\ 0 & 735 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier f4241670	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.60%	78.15%	78.01%	0.57	$\begin{bmatrix} 41 & 17 \\ 9 & 52 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 0ee529e4	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.04%	98.80%	98.79%	0.97	$\begin{bmatrix} 135 & 4 \\ 2 & 358 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 0d8bc91d	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.77%	95.69%	95.63%	0.91	$\begin{bmatrix} 134 & 0 \\ 9 & 66 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 70920273	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	96.62%	96.64%	0.83	$\begin{bmatrix} 31 & 5 \\ 6 & 283 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 38ace6ad	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.48%	99.82%	99.82%	1.0	$\begin{bmatrix} 139 & 0 \\ 1 & 418 \end{bmatrix}$		



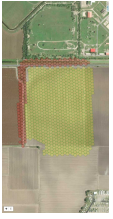
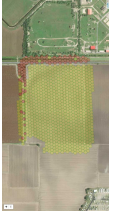






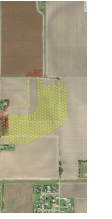
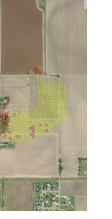


Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier d95c00a0	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.14%	99.14%	0.98	$\begin{bmatrix} 151 & 2 \\ 3 & 423 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 7e639917	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.63%	100.00%	100.00%	1.0	$\begin{bmatrix} 127 & 0 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier e7dd3094	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.70%	98.70%	0.97	$\begin{bmatrix} 171 & 3 \\ 4 & 361 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 9ab9c40f	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.44%	99.30%	99.29%	0.98	$\begin{bmatrix} 54 & 2 \\ 0 & 228 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier d43780c4	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	97.74%	97.74%	0.95	$\begin{bmatrix} 90 & 1 \\ 3 & 83 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 2acb459d	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 86 & 0 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 783f749d	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 216 & 0 \\ 0 & 275 \end{bmatrix}$		















Contrastive Learning with Temperature 0.1 - Parameters					Test Results					
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier fecb6aa	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.65%	99.65%	0.99	$\begin{bmatrix} 196 & 2 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 33862713	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.66%	99.66%	0.98	$\begin{bmatrix} 24 & 0 \\ 1 & 270 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 7a123e6d	15	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.99%	99.03%	0.92	$\begin{bmatrix} 17 & 0 \\ 3 & 277 \end{bmatrix}$		





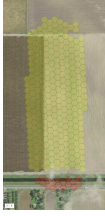
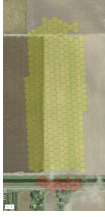
Contrastive Learning with Temperature 0.1 - Parameters				Validation				Test Results			
Description	Width	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted	
Training with test on field 0 and validation on field 1 and test identifier f94387f4	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.74%	96.35%	96.37%	0.91	$\begin{bmatrix} 194 & 11 \\ 17 & 546 \end{bmatrix}$			
Training with test on field 1 and validation on field 2 and test identifier 78a73624	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.12%	96.50%	96.35%	0.87	$\begin{bmatrix} 118 & 29 \\ 2 & 736 \end{bmatrix}$			
Training with test on field 2 and validation on field 3 and test identifier fd9a7cb	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	91.35%	91.34%	0.84	$\begin{bmatrix} 48 & 1 \\ 8 & 47 \end{bmatrix}$			
Training with test on field 3 and validation on field 4 and test identifier 4bf4dc5f	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.60%	99.60%	0.99	$\begin{bmatrix} 137 & 2 \\ 0 & 360 \end{bmatrix}$			
Training with test on field 4 and validation on field 5 and test identifier c281277c	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.10%	98.56%	98.56%	0.97	$\begin{bmatrix} 134 & 0 \\ 3 & 72 \end{bmatrix}$			
Training with test on field 5 and validation on field 6 and test identifier a2275402	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.50%	98.52%	0.93	$\begin{bmatrix} 41 & 1 \\ 4 & 288 \end{bmatrix}$			
Training with test on field 6 and validation on field 7 and test identifier 40bae148	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 139 & 0 \\ 0 & 419 \end{bmatrix}$			



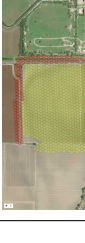
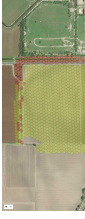






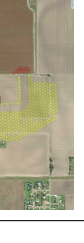
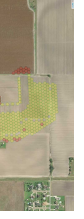
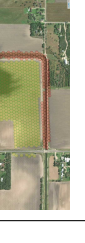
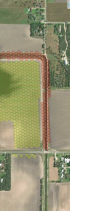
Contrastive Learning with Temperature 0.1 - Parameters				Validation				Test Results			
Description	Width	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted	
Training with test on field 7 and validation on field 8 and test identifier 07bcd95	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	99.83%	1.0	$\begin{bmatrix} 153 & 0 \\ 1 & 425 \end{bmatrix}$			
Training with test on field 8 and validation on field 9 and test identifier f65ad46d	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	99.27%	0.99	$\begin{bmatrix} 125 & 2 \\ 0 & 147 \end{bmatrix}$			
Training with test on field 9 and validation on field 10 and test identifier 7ea1e9a8	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	99.44%	0.99	$\begin{bmatrix} 169 & 0 \\ 3 & 359 \end{bmatrix}$			
Training with test on field 10 and validation on field 11 and test identifier dcea9f8e	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.92%	98.92%	99.65%	0.99	$\begin{bmatrix} 56 & 0 \\ 1 & 227 \end{bmatrix}$			
Training with test on field 11 and validation on field 12 and test identifier f2022295	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	97.84%	0.96	$\begin{bmatrix} 95 & 1 \\ 3 & 86 \end{bmatrix}$			
Training with test on field 12 and validation on field 13 and test identifier ac315be6	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 86 & 0 \\ 0 & 151 \end{bmatrix}$			
Training with test on field 13 and validation on field 14 and test identifier 2a687d87	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	99.80%	1.0	$\begin{bmatrix} 215 & 1 \\ 0 & 275 \end{bmatrix}$			















Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 2fbedf55	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier d60c7a64	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.66%	99.67%	0.98	$\begin{bmatrix} 25 & 0 \\ 1 & 270 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier ae65c78c	30	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.32%	98.42%	0.87	$\begin{bmatrix} 17 & 0 \\ 5 & 276 \end{bmatrix}$		





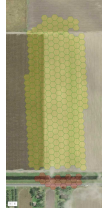
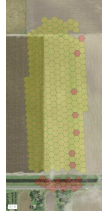
Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier df7429b0	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.45%	97.65%	97.66%	0.94	$\begin{bmatrix} 196 & 8 \\ 10 & 553 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 4d6ba47d	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.44%	93.97%	93.40%	0.77	$\begin{bmatrix} 91 & 53 \\ 0 & 735 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier ad984145	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	80.67%	80.52%	0.62	$\begin{bmatrix} 42 & 16 \\ 7 & 54 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier c89935db	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.20%	98.21%	0.96	$\begin{bmatrix} 139 & 0 \\ 9 & 351 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 3f534144	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.52%	99.52%	0.99	$\begin{bmatrix} 134 & 0 \\ 1 & 74 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 5b97f330	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	93.85%	94.27%	0.75	$\begin{bmatrix} 33 & 3 \\ 17 & 272 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 21e51647	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.82%	99.82%	1.0	$\begin{bmatrix} 138 & 1 \\ 0 & 419 \end{bmatrix}$		



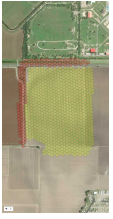
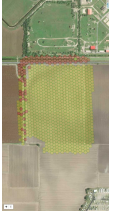






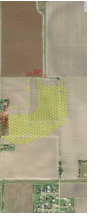
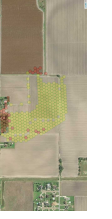


Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 59e80e89	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.83%	99.83%	1.0	$\begin{bmatrix} 153 & 0 \\ 1 & 425 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier a421dbe0	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 127 & 0 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier df7bd166	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.81%	99.81%	1.0	$\begin{bmatrix} 174 & 0 \\ 1 & 364 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier da18cd64	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 56 & 0 \\ 0 & 228 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier e476baeb	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.44%	99.43%	0.99	$\begin{bmatrix} 91 & 0 \\ 1 & 85 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 2e8c715f	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 86 & 0 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier aadedf5e	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.80%	99.80%	1.0	$\begin{bmatrix} 216 & 0 \\ 1 & 274 \end{bmatrix}$		















Contrastive Learning with Temperature 0.1 - Parameters					Test Results					
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 6374ac56	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 64a68ae8	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 24 & 0 \\ 0 & 271 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 74d96058	30	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 17 & 0 \\ 0 & 280 \end{bmatrix}$		







Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier e07d194d	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.95%	96.74%	96.75%	0.92	$\begin{bmatrix} 194 & 11 \\ 14 & 549 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 2af8d927	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.15%	94.69%	94.29%	0.8	$\begin{bmatrix} 101 & 46 \\ 1 & 737 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 5bada20c	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	95.19%	95.19%	0.91	$\begin{bmatrix} 49 & 0 \\ 5 & 50 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 5ca539a7	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.40%	99.40%	0.99	$\begin{bmatrix} 138 & 1 \\ 2 & 358 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier e6d90a11	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.70%	99.52%	99.52%	0.99	$\begin{bmatrix} 134 & 0 \\ 1 & 74 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 877b8d7a	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	96.71%	96.84%	0.87	$\begin{bmatrix} 41 & 1 \\ 10 & 282 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier cbc69fdc	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 139 & 0 \\ 0 & 419 \end{bmatrix}$		



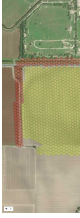
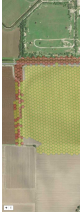










Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 4e5bea14	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.48%	99.48%	0.99	$\begin{bmatrix} 151 & 2 \\ 1 & 425 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier a25803c1	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 127 & 0 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 95f965d5	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.81%	99.81%	1.0	$\begin{bmatrix} 168 & 1 \\ 0 & 362 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier e051fa49	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.65%	99.65%	0.99	$\begin{bmatrix} 55 & 1 \\ 0 & 228 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 5edf1115	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.92%	98.92%	0.98	$\begin{bmatrix} 96 & 0 \\ 2 & 87 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 57515ef1	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 86 & 0 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier d3505c13	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 216 & 0 \\ 0 & 275 \end{bmatrix}$		















Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier bb90cfac	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 8e7f7b3c	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 25 & 0 \\ 0 & 271 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 06b4d372	45	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.87%	97.65%	97.84%	0.83	$\begin{bmatrix} 17 & 0 \\ 7 & 274 \end{bmatrix}$		




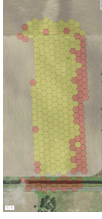
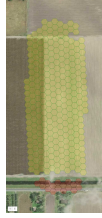

Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 9cad28ae	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.45%	97.39%	97.41%	0.93	$\begin{bmatrix} 199 & 5 \\ 15 & 548 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 47a356ee	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.44%	94.65%	94.26%	0.79	$\begin{bmatrix} 99 & 45 \\ 2 & 733 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier eb7b74dc	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	80.67%	80.62%	0.61	$\begin{bmatrix} 44 & 14 \\ 9 & 52 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 59cfb836	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.00%	99.00%	0.98	$\begin{bmatrix} 137 & 2 \\ 3 & 357 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 6b460668	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.69%	99.52%	99.52%	0.99	$\begin{bmatrix} 134 & 0 \\ 1 & 74 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 482bfd0d	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	96.92%	97.06%	0.86	$\begin{bmatrix} 35 & 1 \\ 9 & 280 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 6ab7ea50	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.82%	99.82%	1.0	$\begin{bmatrix} 139 & 0 \\ 1 & 418 \end{bmatrix}$		



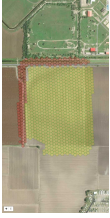
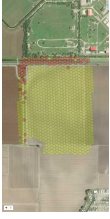










Contrastive Learning with Temperature 0.1 - Parameters				Test Results						
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier bca9517e	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.65%	99.66%	0.99	$\begin{bmatrix} 153 & 0 \\ 2 & 424 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier ee290a28	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 127 & 0 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier fcaaa614	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.63%	99.63%	0.99	$\begin{bmatrix} 174 & 0 \\ 2 & 363 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 590f7014	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.30%	99.30%	0.98	$\begin{bmatrix} 56 & 0 \\ 2 & 226 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 4d3bf12a	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.87%	98.87%	0.98	$\begin{bmatrix} 91 & 0 \\ 2 & 84 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier ac556c9b	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.16%	99.16%	0.98	$\begin{bmatrix} 85 & 1 \\ 1 & 150 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 05d60a85	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.80%	99.80%	1.0	$\begin{bmatrix} 216 & 0 \\ 1 & 274 \end{bmatrix}$		















Contrastive Learning with Temperature 0.1 - Parameters					Test Results					
Description	Width	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 7b117aca	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier e4ec7447	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 24 & 0 \\ 0 & 271 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 026c8ef6	45	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 17 & 0 \\ 0 & 280 \end{bmatrix}$		







Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 50fd0b32	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.95%	95.83%	95.86%	0.9	$\begin{bmatrix} 193 & 12 \\ 20 & 543 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier e70a6c39	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	81.73%	95.59%	95.33%	0.83	$\begin{bmatrix} 109 & 38 \\ 1 & 737 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 01781130	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.20%	77.88%	77.60%	0.56	$\begin{bmatrix} 33 & 16 \\ 7 & 48 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 908049d6	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.04%	99.20%	99.20%	0.98	$\begin{bmatrix} 137 & 2 \\ 2 & 358 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 7d8b076a	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	89.82%	98.09%	98.07%	0.96	$\begin{bmatrix} 134 & 0 \\ 4 & 71 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 0e675b2f	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.28%	88.32%	89.67%	0.65	$\begin{bmatrix} 40 & 2 \\ 37 & 255 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier a7cef93e	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.10%	98.75%	98.73%	0.97	$\begin{bmatrix} 132 & 7 \\ 0 & 419 \end{bmatrix}$		



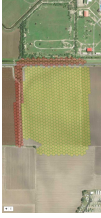
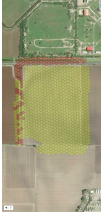










Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier af083313	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.91%	98.27%	98.26%	0.96	$\begin{bmatrix} 145 & 8 \\ 2 & 424 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 138c42f4	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.44%	97.81%	97.81%	0.96	$\begin{bmatrix} 122 & 5 \\ 1 & 146 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier a1c55956	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.18%	97.93%	97.95%	0.95	$\begin{bmatrix} 169 & 0 \\ 11 & 351 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 3a880594	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.84%	96.13%	96.03%	0.87	$\begin{bmatrix} 47 & 9 \\ 2 & 226 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 2b66b444	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	97.84%	97.84%	0.96	$\begin{bmatrix} 95 & 1 \\ 3 & 86 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 1c6c6c25	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.78%	99.58%	99.58%	0.99	$\begin{bmatrix} 86 & 0 \\ 1 & 150 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier da126cc7	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.78%	98.78%	0.98	$\begin{bmatrix} 210 & 6 \\ 0 & 275 \end{bmatrix}$		











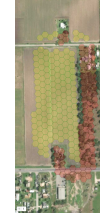



Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 32af6ad6	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.97%	99.65%	99.65%	0.99	$\begin{bmatrix} 198 & 0 \\ 2 & 368 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier cbb57ecf	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.30%	81.08%	84.99%	0.49	$\begin{bmatrix} 25 & 0 \\ 56 & 215 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 1a163767	15	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.96%	83.89%	87.85%	0.47	$\begin{bmatrix} 17 & 0 \\ 48 & 233 \end{bmatrix}$		




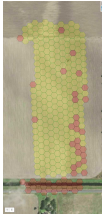
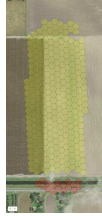
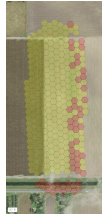
Triplet Loss Siamese with margin 10 - Parameters					Test Results						
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 3c324f5c	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.43%	97.78%	97.80%	0.94	$\begin{bmatrix} 199 & 5 \\ 12 & 551 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier ce52610f	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	68.07%	93.40%	92.73%	0.74	$\begin{bmatrix} 87 & 57 \\ 1 & 734 \end{bmatrix}$		
Training with test on field 3 and validation on field 3 and test identifier 79e60e7e	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.99%	78.15%	77.95%	0.57	$\begin{bmatrix} 40 & 18 \\ 8 & 53 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier eb1237fc	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.26%	96.79%	96.82%	0.92	$\begin{bmatrix} 135 & 4 \\ 12 & 348 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier c3214886	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.77%	89.95%	89.61%	0.78	$\begin{bmatrix} 132 & 2 \\ 19 & 56 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 1f3cc0e0	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.92%	88.92%	90.00%	0.58	$\begin{bmatrix} 29 & 7 \\ 29 & 260 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 43ed2b26	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.45%	98.57%	98.55%	0.96	$\begin{bmatrix} 131 & 8 \\ 0 & 419 \end{bmatrix}$		



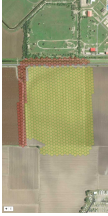
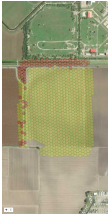










Triplet Loss Siamese with margin 10 - Parameters				Test Results							
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 21a94dfe	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.45%	97.06%	97.03%	0.92	$\begin{bmatrix} 139 & 14 \\ 3 & 423 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier fb2c55ec	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.96%	97.81%	97.81%	0.96	$\begin{bmatrix} 122 & 5 \\ 1 & 146 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 8e9c289c	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.30%	96.85%	96.87%	0.93	$\begin{bmatrix} 170 & 4 \\ 13 & 352 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier e5560764	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.74%	98.94%	98.93%	0.97	$\begin{bmatrix} 53 & 3 \\ 0 & 228 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 778dc310	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	96.05%	96.05%	0.92	$\begin{bmatrix} 87 & 4 \\ 3 & 83 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 74b3d42d	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.19%	98.73%	98.74%	0.97	$\begin{bmatrix} 85 & 1 \\ 2 & 149 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 4864b810	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.82%	98.98%	98.98%	0.98	$\begin{bmatrix} 212 & 4 \\ 1 & 274 \end{bmatrix}$		















Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier cb9e9ae5	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.65%	99.65%	0.99	$\begin{bmatrix} 197 & 1 \\ 1 & 369 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 74746a92	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.33%	95.25%	95.73%	0.77	$\begin{bmatrix} 24 & 0 \\ 14 & 257 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 26eb3386	15	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.70%	94.61%	95.40%	0.7	$\begin{bmatrix} 17 & 0 \\ 16 & 264 \end{bmatrix}$		







Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 963d8178	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.84%	96.22%	96.22%	0.9	$\begin{bmatrix} 190 & 15 \\ 14 & 549 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 76a7c0ab	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.19%	94.92%	94.59%	0.81	$\begin{bmatrix} 105 & 42 \\ 3 & 735 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 7175e81a	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.80%	91.35%	91.35%	0.83	$\begin{bmatrix} 45 & 4 \\ 5 & 50 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier b8a1bac5	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.52%	98.20%	98.20%	0.96	$\begin{bmatrix} 135 & 4 \\ 5 & 355 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 02b63270	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	90.42%	97.61%	97.59%	0.95	$\begin{bmatrix} 134 & 0 \\ 5 & 70 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 51018459	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.28%	84.73%	86.80%	0.59	$\begin{bmatrix} 40 & 2 \\ 49 & 243 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 8eac5301	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.31%	98.92%	98.92%	0.97	$\begin{bmatrix} 133 & 6 \\ 0 & 419 \end{bmatrix}$		



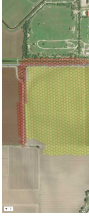
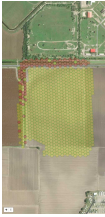








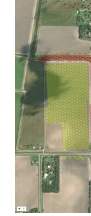

Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 9af0301c	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.18%	98.27%	98.26%	0.96	$\begin{bmatrix} 144 & 9 \\ 1 & 425 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 880a9a45	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.62%	96.35%	96.34%	0.93	$\begin{bmatrix} 117 & 10 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 65512a85	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.30%	97.55%	97.56%	0.94	$\begin{bmatrix} 164 & 5 \\ 8 & 354 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 21d9f018	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.84%	97.89%	97.87%	0.93	$\begin{bmatrix} 52 & 4 \\ 2 & 226 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 5cc66bef	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.58%	97.30%	97.30%	0.95	$\begin{bmatrix} 94 & 2 \\ 3 & 86 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier e57661dc	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.17%	99.58%	99.58%	0.99	$\begin{bmatrix} 85 & 1 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier if069e65	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	97.15%	97.14%	0.94	$\begin{bmatrix} 202 & 14 \\ 0 & 275 \end{bmatrix}$		















Triplet Loss Siamese with margin 10 - Parameters						Test Results					
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier a0f687b2	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.99%	99.65%	99.65%	0.99	$\begin{bmatrix} 196 & 2 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier b8ce3870	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.30%	90.88%	92.24%	0.66	$\begin{bmatrix} 25 & 0 \\ 27 & 244 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier a5d68203	30	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.01%	87.92%	90.61%	0.53	$\begin{bmatrix} 17 & 0 \\ 36 & 245 \end{bmatrix}$		







Triplet Loss Siamese with margin 10 - Parameters				Test Results							
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier f98e1baf	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.61%	97.79%	97.80%	0.94	$\begin{bmatrix} 201 & 4 \\ 13 & 550 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 334417e5	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	91.35%	93.90%	93.37%	0.77	$\begin{bmatrix} 95 & 52 \\ 2 & 736 \end{bmatrix}$		
Training with test on field 3 and validation on field 3 and test identifier 0f9d21c4	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.00%	82.69%	82.63%	0.65	$\begin{bmatrix} 38 & 11 \\ 7 & 48 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier c680f272	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.04%	98.40%	98.39%	0.96	$\begin{bmatrix} 134 & 5 \\ 3 & 357 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 08dc7295	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	90.12%	99.04%	99.04%	0.98	$\begin{bmatrix} 134 & 0 \\ 2 & 73 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 817e006b	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.28%	86.83%	88.32%	0.59	$\begin{bmatrix} 37 & 5 \\ 39 & 253 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 57ad18cb	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.10%	99.28%	99.28%	0.98	$\begin{bmatrix} 136 & 3 \\ 1 & 418 \end{bmatrix}$		


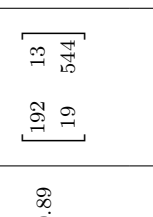

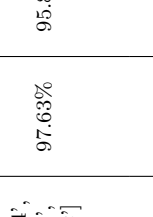
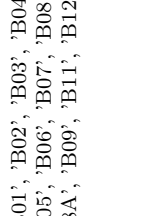

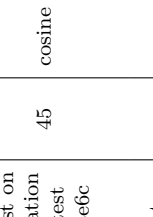
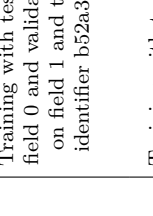

Triplet Loss Siamese with margin 10 - Parameters					Test Results						
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 9394459b	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.54%	97.93%	97.91%	0.95	$\begin{bmatrix} 143 & 10 \\ 2 & 424 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 30bc49c3	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.87%	97.45%	97.44%	0.95	$\begin{bmatrix} 121 & 6 \\ 1 & 146 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier fec6a7bf	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.65%	98.87%	98.87%	0.97	$\begin{bmatrix} 166 & 3 \\ 3 & 359 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 20dc3034	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.30%	98.94%	98.94%	0.97	$\begin{bmatrix} 54 & 2 \\ 1 & 227 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 0e3c3b58	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.58%	95.14%	95.14%	0.9	$\begin{bmatrix} 90 & 6 \\ 3 & 86 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier fc07ab56	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.78%	100.00%	100.00%	1.0	$\begin{bmatrix} 86 & 0 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier d9d5c393	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	98.17%	98.16%	0.96	$\begin{bmatrix} 208 & 8 \\ 1 & 274 \end{bmatrix}$		


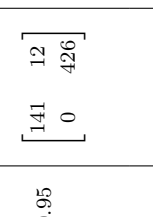

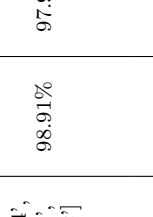
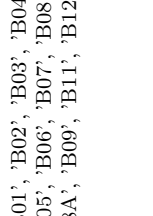

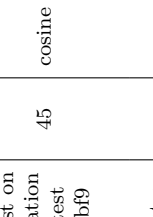
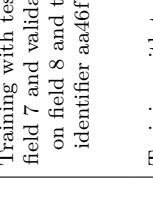

Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 625a5083	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.59%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 973aad25	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.64%	82.77%	86.23%	0.52	$\begin{bmatrix} 25 & 0 \\ 51 & 220 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier db19afec	30	euclidean	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.79%	80.54%	85.55%	0.42	$\begin{bmatrix} 17 & 0 \\ 58 & 223 \end{bmatrix}$		




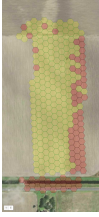
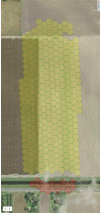
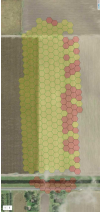
Triplet Loss Siamese with margin 10 - Parameters				Test Results							
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 38cb15a9	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.27%	97.52%	97.52%	0.94	$\begin{bmatrix} 195 & 9 \\ 10 & 553 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 579b026a	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	69.75%	94.08%	93.53%	0.77	$\begin{bmatrix} 92 & 52 \\ 0 & 735 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 52f406fc	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.00%	75.63%	75.27%	0.52	$\begin{bmatrix} 37 & 21 \\ 8 & 53 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 538f0978	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.17%	96.39%	96.38%	0.91	$\begin{bmatrix} 128 & 11 \\ 7 & 353 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 5018a7d4	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	93.54%	94.26%	94.13%	0.88	$\begin{bmatrix} 134 & 0 \\ 12 & 63 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier acb6d038	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.64%	90.46%	91.23%	0.61	$\begin{bmatrix} 29 & 7 \\ 24 & 265 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier 51a72103	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.75%	98.57%	98.55%	0.96	$\begin{bmatrix} 131 & 8 \\ 0 & 419 \end{bmatrix}$		



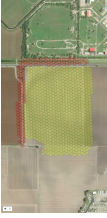
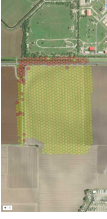










Triplet Loss Siamese with margin 10 - Parameters				Test Results							
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier 51b145ce	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.81%	97.58%	97.55%	0.94	$\begin{bmatrix} 140 & 13 \\ 1 & 425 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 44a545be	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.52%	97.08%	97.07%	0.94	$\begin{bmatrix} 119 & 8 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 6b9d5e4a	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.89%	96.66%	96.68%	0.93	$\begin{bmatrix} 170 & 4 \\ 14 & 351 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 52a29a10	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	98.59%	98.57%	0.96	$\begin{bmatrix} 52 & 4 \\ 0 & 228 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier b0432646	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.16%	97.18%	97.17%	0.94	$\begin{bmatrix} 89 & 2 \\ 3 & 83 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier f3103dc1	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.39%	99.16%	99.16%	0.98	$\begin{bmatrix} 86 & 0 \\ 2 & 149 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier 4ddf8120	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.65%	99.19%	99.18%	0.98	$\begin{bmatrix} 212 & 4 \\ 0 & 275 \end{bmatrix}$		

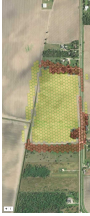





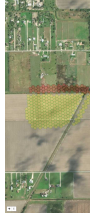



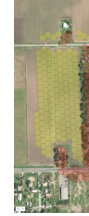
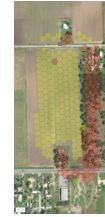


Triplet Loss Siamese with margin 10 - Parameters						Test Results					
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier f90e5f6c	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	99.65%	99.65%	0.99	$\begin{bmatrix} 197 & 1 \\ 1 & 369 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier ce8f61cb	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.66%	95.93%	96.29%	0.8	$\begin{bmatrix} 24 & 0 \\ 12 & 259 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier 7c71fe7d	30	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	91.58%	93.17%	0.61	$\begin{bmatrix} 17 & 0 \\ 25 & 255 \end{bmatrix}$		




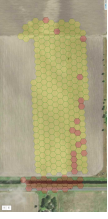
Triplet Loss Siamese with margin 10 - Parameters				Test Results							
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier b52a3e6c	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.63%	95.83%	95.85%	0.89	$\begin{bmatrix} 192 & 13 \\ 19 & 544 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 3211b020	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.19%	94.46%	94.04%	0.79	$\begin{bmatrix} 100 & 47 \\ 2 & 736 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier 1627f734	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.60%	88.46%	88.42%	0.77	$\begin{bmatrix} 41 & 8 \\ 4 & 51 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 2c66766b	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.09%	98.20%	98.19%	0.95	$\begin{bmatrix} 133 & 6 \\ 3 & 357 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier bd1a649f	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	86.83%	97.61%	97.60%	0.95	$\begin{bmatrix} 133 & 1 \\ 4 & 71 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier fd531907	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.28%	85.93%	87.70%	0.6	$\begin{bmatrix} 39 & 3 \\ 44 & 248 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier e06142f6	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.96%	99.10%	99.10%	0.98	$\begin{bmatrix} 135 & 4 \\ 1 & 418 \end{bmatrix}$		

Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier aa46fbf9	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.91%	97.93%	97.90%	0.95	$\begin{bmatrix} 141 & 12 \\ 0 & 426 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier e223db0d	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.25%	97.45%	97.44%	0.95	$\begin{bmatrix} 121 & 6 \\ 1 & 146 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier b76a9bf7	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.65%	98.87%	98.87%	0.97	$\begin{bmatrix} 166 & 3 \\ 3 & 359 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier 24b5849d	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.84%	99.65%	99.65%	0.99	$\begin{bmatrix} 56 & 0 \\ 1 & 227 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier 044baa01	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.58%	97.84%	97.84%	0.96	$\begin{bmatrix} 94 & 2 \\ 2 & 87 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier c1dd9c73	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.37%	99.58%	99.58%	0.99	$\begin{bmatrix} 85 & 1 \\ 0 & 151 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier bfc2e297	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	97.96%	97.96%	0.96	$\begin{bmatrix} 207 & 9 \\ 1 & 274 \end{bmatrix}$		

Triplet Loss Siamese with margin 10 - Parameters							Test Results				
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 09f05cb2	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	100.00%	100.00%	100.00%	1.0	$\begin{bmatrix} 198 & 0 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier 56c9cbe7	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	92.95%	81.08%	84.99%	0.49	$\begin{bmatrix} 25 & 0 \\ 56 & 215 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier ebb371e	45	cosine	season	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.70%	84.23%	88.08%	0.47	$\begin{bmatrix} 17 & 0 \\ 47 & 234 \end{bmatrix}$		

Triplet Loss Siamese with margin 10 - Parameters				Test Results						
Description	Width	Loss	Sampling	Bands	Validation			Test Results		
					Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 0 and validation on field 1 and test identifier 99105afc	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	95.22%	98.44%	0.96	$\begin{bmatrix} 199 & 5 \\ 7 & 556 \end{bmatrix}$		
Training with test on field 1 and validation on field 2 and test identifier 98cdc349	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	70.59%	93.17%	0.73	$\begin{bmatrix} 86 & 58 \\ 2 & 733 \end{bmatrix}$		
Training with test on field 2 and validation on field 3 and test identifier e3f45067	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.60%	77.31%	0.55	$\begin{bmatrix} 39 & 19 \\ 8 & 53 \end{bmatrix}$		
Training with test on field 3 and validation on field 4 and test identifier 598dd532	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	96.17%	97.60%	0.94	$\begin{bmatrix} 133 & 6 \\ 6 & 354 \end{bmatrix}$		
Training with test on field 4 and validation on field 5 and test identifier 554e6167	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	94.46%	91.87%	0.83	$\begin{bmatrix} 133 & 1 \\ 16 & 59 \end{bmatrix}$		
Training with test on field 5 and validation on field 6 and test identifier 6f414683	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.82%	89.54%	0.59	$\begin{bmatrix} 29 & 7 \\ 27 & 262 \end{bmatrix}$		
Training with test on field 6 and validation on field 7 and test identifier b01f72c1	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.96%	99.28%	0.98	$\begin{bmatrix} 135 & 4 \\ 0 & 419 \end{bmatrix}$		

Triplet Loss Siamese with margin 10 - Parameters				Test Results							
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 7 and validation on field 8 and test identifier e2f5a4bc	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.54%	97.41%	97.38%	0.93	$\begin{bmatrix} 141 & 12 \\ 3 & 423 \end{bmatrix}$		
Training with test on field 8 and validation on field 9 and test identifier 3df4c930	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	97.96%	97.08%	97.07%	0.94	$\begin{bmatrix} 119 & 8 \\ 0 & 147 \end{bmatrix}$		
Training with test on field 9 and validation on field 10 and test identifier 13437624	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.94%	98.52%	98.52%	0.97	$\begin{bmatrix} 171 & 3 \\ 5 & 360 \end{bmatrix}$		
Training with test on field 10 and validation on field 11 and test identifier bd94e3b0	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	98.24%	98.21%	0.94	$\begin{bmatrix} 51 & 5 \\ 0 & 228 \end{bmatrix}$		
Training with test on field 11 and validation on field 12 and test identifier e37b82c5	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.58%	97.74%	97.74%	0.95	$\begin{bmatrix} 89 & 2 \\ 2 & 84 \end{bmatrix}$		
Training with test on field 12 and validation on field 13 and test identifier 962edc57	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.80%	98.31%	98.31%	0.96	$\begin{bmatrix} 84 & 2 \\ 2 & 149 \end{bmatrix}$		
Training with test on field 13 and validation on field 14 and test identifier b76bb657	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.82%	97.96%	97.96%	0.96	$\begin{bmatrix} 207 & 9 \\ 1 & 274 \end{bmatrix}$		

Triplet Loss Siamese with margin 10 - Parameters						Test Results					
Description	Width	Loss	Sampling	Bands	Validation Accuracy	Accuracy	F1 Score	MCC	Confusion Matrix	Actual	Predicted
Training with test on field 14 and validation on field 15 and test identifier 83a9a419	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.32%	99.82%	99.82%	1.0	$\begin{bmatrix} 197 & 1 \\ 0 & 370 \end{bmatrix}$		
Training with test on field 15 and validation on field 16 and test identifier d223e78d	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	99.66%	94.58%	95.17%	0.75	$\begin{bmatrix} 24 & 0 \\ 16 & 255 \end{bmatrix}$		
Training with test on field 16 and validation on field 0 and test identifier b2c6c5a0	45	cosine	shuffle	['B01', 'B02', 'B03', 'B04', 'B05', 'B06', 'B07', 'B08', 'B8A', 'B09', 'B11', 'B12']	98.31%	94.28%	95.14%	0.69	$\begin{bmatrix} 17 & 0 \\ 17 & 263 \end{bmatrix}$	