

UNIVERSITY OF BAHRAIN
College of Information Technology



Deep Clustering Techniques for Plant Disease Discovery

A Thesis Submitted in Partial Fulfilment of the Requirements for the
Doctor of Philosophy Degree in Computing and Information Sciences

Submitted By

Alya Sayed Husain Ahmed Husain AlKameli

19990221

Supervised By

Dr. Riadh Ksantini

(Associate Professor)

Co-Supervisor

Dr. Hesham Al-Ammal

(Assistant Professor)

University of Bahrain

November 2024



استمارة إيداع البحث

عالية السيد حسين أحمد الكامل			اسم الباحث:
تقنيات التجميع العميق لاكتشاف الأمراض النباتية			عنوان البحث:
2024			سنة النشر:
هندسة علوم الحاسوب	القسم:	كلية تقنية المعلومات	الكلية:
34172222	رقم الاتصال:	aakameli@uob.edu.bh	البريد الإلكتروني:

إقرار الباحث بالموافقة على الشروط التالية:

يحق لجامعة البحرين إتاحة وتوفير الرسالة أو الأطروحة المذكورة أعلاه في مكتبة الجامعة أو من خلال موقع الإسهامات البحثية للجامعة أو على نطاق أوسع في الخارج؛ كما ويمكن إعادة طباعة الرسالة إذا اقتضت الحاجة لذلك أو رقمتها مع إتاحتها على شبكة الأنترنت وذلك حسب تقدير إدارة الجامعة.

تنطبق هذه الشروط على جميع النسخ بما في ذلك النسخ الإلكترونية

تعتبر هذه الرسالة أو الأطروحة من المواد المحمية والتي تخضع إلى حقوق النشر والتأليف، ولا يجوز نشر أي اقتباس من الرسالة بدون توثيق المصدر .

توقيع الباحث: عالية السيد حسين الكامل التاريخ: 12 May, 2024

DECLARATION

I declare that this work is the result of my own investigation and that it has not already been accepted in substance for any degree, nor is it currently submitted for any degree.

Signed:

Date: 12/05/2024

Alia Sayed Husain Ahmed AlKameli (Candidate)

The Thesis defense committee considers the thesis titled:

Deep Clustering Techniques for Plant Disease Discovery

Submitted by Alia Sayed Husain Ahmed AlKameli to the College of Information Technology is satisfactory and acceptable for the Doctorate of Philosophy Degree in Computing and Information Sciences.

The Defense Committee:

Dr. Riadh Ksantini (Supervisor), Department of Computer Science,
University of Bahrain

Dr. Hesham Al-Ammal (Co-Supervisor), Department of Computer
Science, University of Bahrain

XXX (Member), University of xxx

XXX (Member), Department of xxx, University of Bahrain

ABSTRACT

Plant disease detection is essential for maintaining food security and agricultural output. The complexity and diversity of plant disease datasets are often too much for traditional methods to handle, so new ideas are needed to improve disease discovery processes. In order to better control and monitor plant diseases, this thesis looks at the use of deep clustering approaches for disease discovery. It does this by utilizing the capabilities of cutting-edge machine learning techniques.

Through a comprehensive exploration of deep clustering algorithms, including dynamic autoencoders (DynAEs), autoencoder with KMeans, First Integer Neighbor Clustering Hierarchy (FINCH), Variational Deep Embedding (VADE), deep embedded clustering (DEC), and Improved Deep Embedded Clustering (IDEC), the thesis examines their efficacy in accurately categorizing plant diseases. Our examinations illustrate that when compared to all the other five deep clustering models, DynAE continues to lead to better results in terms of ACC and NMI. Comparative study shed a light on the advantages and disadvantages of each of the mentioned methods, offering insightful information on how well they function in actual situations.

The key findings of this thesis reveal that DynAE significantly outperformed other state-of-the-art deep clustering methods, such as autoencoder with K-Means, FINCH, VADE, DEC, and IDEC, across four distinct plant disease datasets—potato, apple, corn, and grape. The performance was evaluated using accuracy (ACC) and normalized mutual information (NMI) as metrics. Remarkably, DynAE achieved the highest accuracy of 72.21% on the potato dataset, indicating its superior ability to correctly cluster the data points compared to other methods. Additionally, DynAE recorded the highest NMI of 63.37% on the corn dataset, demonstrating its effectiveness in maintaining the mutual information between the predicted and true clusters. These results underscore DynAE’s robustness and potential as a leading approach for clustering tasks in plant disease detection. Following the analysis of DynAE’s overall performance, an important enhancement in this work was the substitution of the K-means clustering algorithm with the Gaussian Mixture Model (GMM) in the DynAE framework. The original DynAE method relied on K-means for clustering, but this thesis demonstrates that GMM significantly improves clustering accuracy and robustness. DynAE with GMM achieved an even higher accuracy of 78.99% on the corn dataset, along with a slightly improved NMI of 63.88% on the same dataset. This modification leads to superior clustering performance, with DynAE using GMM outperforming not only the original DynAE model with K-means, but also other deep clustering methods.

Contents

APPROVAL	A
DECLARATION	B
ABSTRACT	C
TABLE OF CONTENTS	D
LIST OF FIGURES	H
LIST OF TABLES	J
ACKNOWLEDGMENTS	L
DEDICATION	M
PUBLICATIONS	N
LIST OF ABBREVIATIONS	O
1 Introduction	1
1.1 Overview	1
1.2 Background	2
1.3 The Problem Statement	4
1.3.1 Research Aim	6
1.4 Research Significance	7
1.4.1 Research Contribution	8
1.5 Research Questions	8
1.6 Research Objectives	9
1.7 Research Settings	10
1.7.1 Thesis Structure	10

1.8	Summary	12
2	Literature Review	13
2.1	Overview	13
2.2	Background	14
2.3	Methodology	21
2.3.1	Related Surveys	22
2.4	Machine Learning in Agriculture	23
2.5	Deep Learning in Agriculture	27
2.6	Deep Clustering Algorithms	38
2.7	Deep Clustering for Plant Disease Detection	43
2.8	Summary	50
3	Methods and Procedures	52
3.1	Overview	52
3.2	The Challenge	56
3.2.1	Feature Randomness	58
3.2.2	Feature Drift	59
3.3	Proposed Dynamic Autoencoder Model	62
3.3.1	Pretraining	64
3.3.2	Clustering phase	66
3.3.3	Optimization	71
3.3.4	The main contribution	71
3.4	Baseline Clustering Methods	74
3.5	Summary	78
4	Empirical Results and Discussion	79
4.1	Overview	79
4.2	Experimental Settings	80
4.2.1	Baseline and dataset	81

4.2.2	Evaluation Metrics	84
4.2.3	Implementation and platform	90
4.3	Experimental Results	90
4.4	Discussion	101
4.5	Threats to Validity	104
4.6	Summary	106
5	Conclusion and Recommendations	107
5.1	Overview	107
5.2	Recommendations and Future Work	108
5.2.1	General Recommendations related to PDD	109
5.2.2	CEA in the GCC	109
5.2.3	Deep Clustering for PDD	109
5.3	The Final Note	110
	References	112
	Abstract (in Arabic)	

List of Figures

1.1	The Structure of the Thesis	11
2.1	CEA Framework Overview	15
2.2	Number of publications in ML applications in Agriculture	19
2.3	A comparison of the number of publications conducted about the application of ML and DL in agriculture	19
2.4	Agricultural applications utilizing ML in the surveyed sample	26
2.5	A generic architecture of traditional ANN and CNN	27
3.1	Overall framework of the first deep clustering category: embedded learning is conducted prior to clustering	53
3.2	Overall framework of the second deep clustering category: simultaneously conducting embedded learning and clustering	54
3.3	The network topology that most autoencoder-based clustering techniques share	55
3.4	dynAE Architecture	63
3.5	Using DynAE clustering architecture to detect plant diseases	67

4.1	Plant leaf diseases Dataset (Plant Village)	82
4.2	ACC between DynAE and IDEC on Grape	95
4.3	NMI between DynAE and IDEC on Grape	95
4.4	Learning dynamics of DynAE (KMeans) on Potato Dataset	96
4.5	Learning dynamics of DynAE (GMM) on Potato . .	97
4.6	t-SNE visualization of the latent representations of DynAE	99
4.7	Measured intrinsic dimension on Potato (Feature Twist)	100

List of Tables

2.1	Secondary research about ML applications in agriculture in the last 5 years	20
2.2	Conventional ML applications in agriculture	24
2.3	DL applications in agriculture	31
2.4	Related works comparison	44
2.5	Related secondary research work	51
3.1	Key differences between GMM and K-means	72
4.1	Plant leaf diseases dataset	81
4.2	Plant leaf diseases dataset	83
4.3	Potato Dataset - DEC, IDEC and DynAE Hyperparameters	86
4.4	Apple Dataset - DEC, IDEC and DynAE Hyperparameters	87
4.5	Corn Dataset - DEC, IDEC and DynAE Hyperparameters	88
4.6	Grape Dataset - DEC, IDEC and DynAE Hyperparameters	89
4.7	Comparison of the clustering performances in terms of ACC and NMI. Best method in bold	91
4.8	Ablation study of DynAE (Kmeans) / DynAE (GMM) : p_i indicates the i th phase	93

4.9	Comparison of the execution times (in seconds) of different deep clustering approaches	94
-----	---	----

ACKNOWLEDGMENTS

I extend my heartfelt appreciation to my supervisors, Dr. Riadh Ksantini and Dr. Hesham Al-Ammal, for their invaluable support, assistance, and unwavering motivation throughout my research journey and the composition of this thesis. Their guidance has been a constant source of inspiration, providing crucial direction at every step of the process. I am sincerely grateful for their expertise, encouragement, and dedication, which have significantly contributed to the completion of this work.

Furthermore, I extend my heartfelt thanks to the Dean of College of Information Technology, Dr. Hessa Al-junaid, and the Chairperson of Computer Science Department, Dr. Abdulla Alasaadi and the former Chairperson of Computer Science Department, Dr. Ahmed Fahad. Their consistent support and boundless encouragement have transformed what could have been a challenging research path into a fulfilling and rewarding expedition. Their leadership and belief in my capabilities have been instrumental, contributing significantly to the accomplishment of this academic endeavor.

DEDICATION

To my cherished daughters, Basma and Sarah, whose presence in my life serves as the unwavering source of motivation propelling me forward in the pursuit of this work. In moments of challenge, it is the thought of your smiling faces that has fueled my determination, reminding me of the profound significance of the work at hand. As I embark on this academic journey, my ultimate aspiration is for the three of us to collectively embody resilience, intelligence, and an unwavering bond of love. May this dedication not only stand as a testament to the enduring connection we share but also serve as a reminder of the values that unite us on a shared path of growth.

PUBLICATIONS

This thesis is based on multiple research papers. Previously published papers listed below:

1. AlKameli, Alia, and Mustafa Hammad. "Automatic learning in agriculture: a survey." International Journal Of Computing and Digital System (2021).
2. AlKameli, A. S., and W. M. ElMedany. "IoT-enabled controlled environment agriculture." (2021): 296-301.

LIST OF ABBREVIATIONS

ACC	Accuracy
ACM	Association for Computing Machinery
AE	Autoencoders
ANN	Artificial Neural Networks
ADEC	Adversarial Deep Embedded Clustering
AI	Artificial Intelligence
BN	Bayesian Network
CC	Cloud Computing
CEA	Controlled Environment Agriculture
CEPPS	Controlled Environment Plant Production System
CNN	Convolutional Neural Networks
DCN	Deep Clustering Network
DEC	Deep Embedding for Clustering
DenseNet	Densely Connected Convolutional Networks
DEPICT	Deep Embedded Regularized Clustering
DFANet	Deep Feature Aggregation Net
DL	Deep learning
DNN	Deep Neural Networks

DT	Decision Trees
DynAE	Dynamic Autoencoder
ELBO	Evidence Lower BOund
ELISAs	Enzyme-linked immunosorbent assays
FCM	Flow Cytometry
FINCH	First Integer Neighbor Clustering Hierarchy
FISH	Fluorescence in situ hybridization
GA	Genetic Algorithm
GCC	Gulf Cooperation Council
GC-MS	Gas Chromatography-Mass Spectrometry
GMM	Gaussian Mixture Model
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
GRU	Gated Recurrent Units
HSI	Hyperspectral Imaging
IDEC	Improved Deep Embedded Clustering
IEEE	Institute of Electrical and Electronics Engineers
IF	Immunofluorescence
IoT	Internet of Things

JULE	Joint Unsupervised LEarning
KNN	K-Nearest Neighbors
MI	Mutual Information
ML	Machine Learning
MLR	Multiple Linear Regression
MSE	Mean Squared Error
NMI	Normalized Mutual Information
PA	Precision Agriculture
PCR	Polymerase Chain Reactions
PDD	Plant Disease Discovery
ResNet	Residual Networks
RGB	Red, Green and Blue
RNN	Recurrent Neural Network
RSI	Remote Sensing Images
SGD	Stochastic Gradient Descent
SLR	Systematic Literature Review
SVM	Support Vector Machine
SVR	Support Vector Regression
t-SNE	t-distributed Stochastic Neighbor Embedding

UAVs	Unmanned Aerial Vehicles
VaDE	Variational Deep Embedding
VAE	Variational Autoencoder
VGG	Visual Geometry Group
VRT	Variable Rate Technologies
WSAN	Wireless Sensor and Actuator Network

Chapter 1

Introduction

1.1 Overview

Most of the solutions for agricultural Machine Learning (ML) decision-making problems are using supervised approaches. In supervised learning, the machine needs to be trained on samples of inputs and outputs to support decision-making. In the fast changing environment of ML, deep clustering has emerged as a potent and adaptable strategy for addressing unsupervised learning difficulties. Deep clustering is the use of deep neural networks to automatically identify hidden patterns and structures in unlabeled data.

Traditional clustering approaches rely on handmade features and established similarity measures, but they struggle with high-dimensional and unstructured data. Deep clustering, on the other hand, takes advantage of the expressive capacity of Deep Learning (DL) architectures to automatically generate elaborate representations from raw input data. This not only minimizes the need for human feature engineering, but it also allows for the

identification of complex correlations that would be difficult to detect using traditional approaches. Deep clustering has several applications, including image and voice processing, natural language comprehension, and more. Its capacity to detect latent features in various datasets makes it especially useful in situations when labeled data is sparse or expensive to gather.

In this introductory chapter we lay down the background for this work in section 1.2. This is followed by discussing the problem statement and the research aim in section 1.3. Following that we shed a light on the research Significance and our contribution in section 1.4. The research questions and research objectives are discussed in sections 1.5 and 1.6. The research settings and thesis structure are provided in section 1.7. Finally, the chapter ends with a summary in section 1.8.

1.2 Background

Agriculture plays a pivotal role in the growth of any nation. Nowadays, with the advancements of information technologies, big data are generated and executed using faster computing techniques. Agriculture has a big impact on the living standards of people. It has many important roles; one of its roles is sustaining people's lives by supplying them with food needed to continue thriving. The supply of food for people and fodder for livestock depends on the agricultural sector. Generally speaking, agriculture is an important sector to make sure a stable chain of food supply exists and to ensure food security and machine learning techniques have been used to automate and improve agricultural activities for a long time..

Some countries have very extreme resources' constraints. For instance, the Gulf Cooperation Council (GCC) countries have environments that are naturally unsuitable for farm production. The three main constraints related to the agricultural sector in this region are weather, water and arable land.

In general, GCC countries have longer hot summers and shorter cool winters. For instance, Bahrain has an arid to an extremely arid environment, in which the average temperature exceeds 35-40 °C in the summer season (May to October) (Bayram, Al-Qahtani, Saffouri, & Koç, 2017). On the other hand, the main source of water in the region is groundwater which is mostly nonrenewable due to the limitation of rainfall which feeds the groundwater. Rainfall in this region is rare, with an average annual rainfall that ranges between 1-100 mm (Abdelbaki, 2016). Moreover, according to Pirani and Arafat (Pirani & Arafat, 2016), less than 2% of GCC total land area is cultivable lands. In some countries, outsourcing agricultural production by acquiring lands abroad is considered a solution to face the extreme resource constraints. However, this solution does not assure self-sufficiency, efficiency and long-term sustainability. A possible practical solution for this problem is using Controlled Environment Agriculture (CEA) technologies.

CEA is about using technology to produce food, such that growing conditions are optimized and protection is provided throughout the crop lifecycle. Usually, in CEA, agricultural activities are held within an indoor farm or greenhouse. CEA methods include hydroponics (Jones Jr, 2016), aeroponics (Lakkireddy, Kasturi, & KRS, 2018), aquaponics (Rakocy, Masser, Losordo, et al., 2016) and aquaculture (Lucas, Southgate, & Tucker, 2019). Generally, CEA focuses on optimizing resources such as water, energy, space and labor. For instance, vertical farms (Benke & Tomkins, 2017) are used to optimize space and hydroponics (Jones Jr, 2016) are used to optimize water and nutrients.

CEA is already in use in several farms in GCC countries. Mostly these farms are inside a building or a greenhouse. Most of these farms are utilizing modern soil-less agricultural methods such as hydroponics. Some are even using vertical farming to efficiently utilize the space of the greenhouse. However, there is a shortage of studies in the literature discussing the

application of CEA technologies in the GCC region. This might be justified by the fact that GCC countries are not agricultural. Though, greenhouse and indoor farms are used for a long time in almost all the GCC countries. This paper aims to focus on improving the monitoring and control of indoor/greenhouse farming activities.

Greenhouse farming is different from open-field farming. In a greenhouse, the variables related to farming can be controlled more efficiently, fully or partially. However, the optimization of crop growth and all CEA variables such as temperature, humidity, light, nutrients, acidity and pests is a multi objective problem with the presence of some uncertainty.

To come up with optimal solutions and unbiased opinion about the climate condition of a greenhouse, various measurements are required at many spots in the greenhouse. To accurately and precisely measure the different variables, a distributed sensing system is required. Using Wireless Sensor and Actuator Network (WSAN), developing a distributed monitoring and controlling system is achievable (Bressan et al., 2010). A WSAN consists of a group of battery-powered sensing nodes that gather information about the environment as well as actuators, such as servos or motors, which communicate with them. All the elements in a WSAN interact wirelessly (Raza, Faheem, & Guenes, 2019). The collected information from sensing nodes can also be transmitted wirelessly to a central base station which can store the data for future processing. Moreover, to make the best decision based on the collected data, AI can be used to learn rules directly from the real-life data instead of manually programming them.

1.3 The Problem Statement

Agricultural activities have a significant impact on the development of human life and prosperity. Agriculture is affected greatly by plant diseases,

which pose a serious threat to agricultural production. To control plant diseases and produce quality crops, accurate and rapid identification of pathogens is essential.

Deep Learning (DL) has recently demonstrated considerable potential in handling plant disease assessment problems. It is preferable and most suitable when dealing with high-dimensional, large-scale, and high-semantic data. For high-dimensional data, neural networks are attractive due to their ability to project samples into lower-dimensional spaces. In addition, neural networks can evade time and memory bottlenecks by applying mini-batch Stochastic Gradient Descent (SGD). Thus, large-scale datasets can be processed using multi-layer architecture. Moreover, biological datasets naturally complement deep neural networks' hierarchical architecture. It is, therefore, appropriate to process a high-semantic plant disease dataset using a multi-layer architecture.

In this work, we approach the plant disease assessment problem as a deep clustering problem. We use datasets consisting of several plant diseases and aim to cluster the instances into groups of plant diseases. In Machine Learning (ML), clustering has been researched for decades. Several methods have been presented to segment the data based on fundamental similarities while avoiding the need for labeled data.

The majority of known deep clustering algorithms use an auto-encoding design (Xie, Girshick, & Farhadi, 2016; X. Guo, Gao, Liu, & Yin, 2017; B. Yang, Fu, Sidiropoulos, & Hong, 2017; Tzoreff, Kogan, & Choukroun, 2018; S. A. Shah & Koltun, 2018). Other systems rely on an encoding network in the absence of a decoder (Haeusser, Plapp, Golkov, Aljalbout, & Cremers, 2018; J. Yang, Parikh, & Batra, 2016; Chang, Wang, Meng, Xiang, & Pan, 2017; Caron, Bojanowski, Joulin, & Douze, 2018; Hu, Miyato, Tokui, Matsumoto, & Sugiyama, 2017). The encoder training, in the first technique, is regularized by a reconstruction cost. As a result, the decoder

is employed to mitigate the effect of training with false labels. The latter technique involves two stages: clustering the embedded features using a classical clustering algorithm and updating the encoding weights using the following pseudo-labels as a supervisory signal. However, by removing the decoder network and training the encoder on hypothetical similarities, the space topology might get frequently distorted. The encoder can potentially create arbitrary features that do not accurately represent the true discriminative properties. This is known as Feature Randomness (Mrabah, Bouguessa, & Ksantini, 2020).

To limit the effect of Feature Randomness, autoencoders aim to recreate important features without inducing any randomness. However, autoencoders are challenging due to the natural trade-off resulting from combining clustering and reconstruction. In autoencoders, the clustering objective function focuses on removing non-discriminatory features. On the other hand, the reconstruction cost function aims to preserve all information regardless of its discrimination nature. For example, an image of a potato leaf and one of a tomato leaf would both include numerous background dirt pixels and few showing the actual objects. In this situation, the reconstruction would encourage encoding information, which is unimportant for clustering. (i.e., the color of the soil). This is known as Feature Drift.

1.3.1 Research Aim

The main aim of this work is to propose a model that deal with the shortcomings of other autoencoder-based deep clustering algorithms and to utilize it on a Plant Disease Discovery (PDD) problem.

1.4 Research Significance

Plant diseases impact agriculture significantly and pose an immense threat to agricultural production. To control plant diseases and produce quality crops, accurate and rapid identification of pathogens is essential. There are several laboratory-based plant disease detection methods available. Plant pathologists' approaches include polymerase chain reactions (PCR), immunofluorescence (IF), fluorescence in situ hybridization (FISH), enzyme-linked immunosorbent assays (ELISAs), flow cytometry (FCM), and gas chromatography-mass spectrometry (GC-MS). There are also indirect methods to detect plant diseases, such as thermography, fluorescence imaging, and hyperspectral techniques (Fang & Ramasamy, 2015). Also, many labs use ML algorithms to analyze imaging and sensory data and assess plant diseases accordingly. Recently, the agriculture research industry made great use of advances in the DL field, and lots of research studies include some DL techniques to identify plant diseases. This research focuses on imaging technologies and deep learning applications to detect plant diseases.

According to Barbedo (J. G. Barbedo, 2022), in plant disease assessment, images, and deep learning, together, can be used in three main tasks: disease classification, detection, and severity estimation. A classification task involves determining which disease is responsible for a symptom or sign. Usually, the disease detection task is approached as a binary classification task that categorizes everything other than the disease of interest as a second class. After the classification or identification, the disease symptom severity estimation attempts to provide objective measurement of the extent of symptoms observed, for instance, the percentage of leaf area. To the three categories discussed by Barbedo (J. G. Barbedo, 2022), we add a fourth category: plant disease categorization using clustering techniques. After reviewing the literature, we found that researchers seldom use clustering or deep clustering algorithms for categorizing plant diseases.

1.4.1 Research Contribution

In this work, we propose using a model that has a dynamic loss function to deal with the shortcomings of other autoencoder-based deep clustering algorithms. This function gradually and smoothly dispenses with the reconstruction loss in favor of a construction one. Hence, our model deals with the challenging trade-off between Feature Randomness and Feature Drift.

1.5 Research Questions

In this study, we are aiming to develop a novel framework for plant disease categorization using state-of-the-art deep clustering algorithms. We structured the research questions to help achieve our research objectives. The research questions are as follow:

Deep Clustering for Plant Disease Discovery

RQ1. How can we utilize ML and DL in improving agricultural activities?

RQ2. What are the current trends in plant disease discovery in general and in plant disease clustering?

RQ3. What are the most appropriate clustering methods that can be used to categorize plant disease?

Dynamic Autoencoder for Plant Disease Discovery

RQ4. Can the Dynamic Autoencoder (DynAE) be used to improve plant disease categorization and can it outperform modern clustering models on real benchmark plant disease image datasets?

RQ5. How can we alleviate the two problems of feature drift and feature randomness in plant disease categorization?

RQ6. Can we overcome the clustering-reconstruction trade-off with a dynamic loss function?

RQ7. How can we improve the DynAE to better solve the plant disease discovery problem?

1.6 Research Objectives

The main aim, in this work, is to use more accurate imaging data such as Hyperspectral Imaging (HSI) and Multispectral Imaging and ML techniques to cluster plant diseases.

Originally, the research aimed to create an HSI local dataset to be used for plant disease discovery and deep learning studies. However, an unexpected global pandemic disrupted the project, rendering the initial objective unattainable. In response, the thesis swiftly adapted to explore the use of deep clustering for plant disease discovery using pre-existing RGB imaging dataset. The main objectives of this research project are:

RO1. To utilize DL in improving agricultural activities.

RO2. To investigate current trends in plant disease discovery.

RO3. To investigate the most appropriate clustering methods to categorize plant disease.

RO4. To create a novel model for plant disease discovery that combines the current state of the art DL algorithms with the use of an imaging dataset.

RO5. To examine the use of DynAE to improve plant disease categorization.

RO6. To Investigate the effectiveness of DynAE in handling feature drift and feature randomness.

1.7 Research Settings

This research is considered a quantitative research that follows the empirical paradigm and the scientific method. The central problem of this research study is a deep clustering problem in which we cluster plants as healthy or diseased plants based on leaves images dataset. We intend to solve this problem using deep clustering algorithms. We use pre-existing non-local plant disease datasets. The project require Computing device with powerful GPU for deep learning model training. We expect to gain better understanding of deep clustering techniques and its applications in plant disease discovery.

1.7.1 Thesis Structure

The rest of this work is structured as follows. The next chapter provides a detailed literature review of previous studies related to this work. Chapter 3 discusses the methods and procedures used in this study, focusing on the proposed DynAE model. After that, in chapter 4, we present the experimental settings and we compare the results obtained by the proposed model with existing methodologies and provide graphical representations to

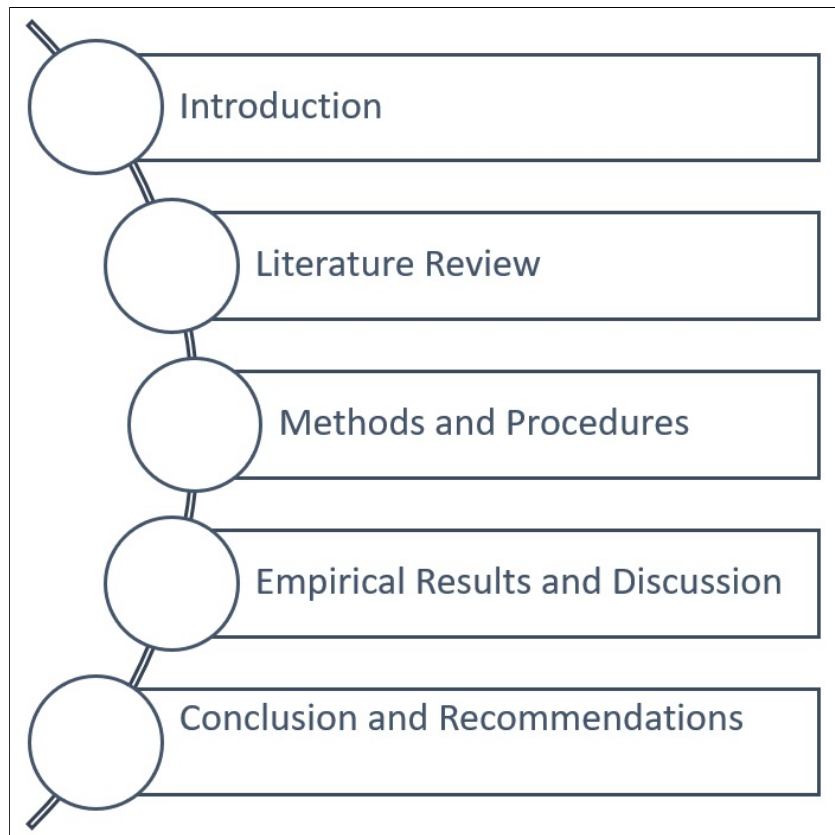


Figure 1.1: The Structure of the Thesis

illustrate its effectiveness. Finally, chapter 5 emphasize the model's effectiveness and discuss the possible future directions of the research. Figure 1.1 illustrates the structure of the thesis report.

1.8 Summary

The introduction chapter serves as a foundation for the subsequent chapters, guiding readers through the motivation, objectives, and significance of the research. By highlighting the critical role of deep clustering in addressing challenges within plant disease discovery, the thesis sets the tone for an in-depth exploration of how cutting-edge technology can contribute to the advancement of sustainable crop management and agricultural practices. The following chapter explores current research, methodologies, and advancements in using deep learning, namely clustering algorithms, for plant disease detection and diagnosis.

Chapter 2

Literature Review

2.1 Overview

In recent years, the agricultural landscape has seen a paradigm change in the pursuit of sustainable crop management, with technology playing an important part in resolving the issues faced by plant diseases. Deep learning has opened up new vistas in plant pathology, providing unique methods for the early identification and precise categorization of crop-related illnesses. This literature review examines the present status of research, approaches, and advances in the use of deep learning, particularly deep clustering algorithms for plant disease discovery and diagnosis.

In this chapter our aim is to survey current research efforts to explore the possibility of using Deep Learning (DL) techniques to propose an advanced Artificial Intelligence (AI) solution to improve agricultural decision-making. In the next section, we synthesize current information about different technologies used to improve different agricultural activities. The

methodology of the survey is discussed in section 2.3. Followed by a highlight of the research efforts using Machine Learning (ML) in agriculture in section 2.4. Then, in section 2.5, we focus on the application of DL in agriculture. Following that, we shed a light on the state-of-the-art deep clustering algorithms in section 2.6. We delve into a review of literature on Plant Disease Discovery (PDD) using deep clustering in section 2.7. Finally, the chapter ends with a summary in section 2.8.

2.2 Background

In recent times, a wide range of technologies have been used to make farming more efficient. For instance, farmers use Global Positioning System (GPS) and Global Navigation Satellite System (GNSS) (Murugan, Garg, & Singh, 2017) to optimize farming activities such as irrigation, harvesting, fertilizing and pest control. Other technologies used to support the monitoring and control of agricultural activities includeing Wireless Sensor and Actuator Network (WSAN) (Anisi, Abdul-Salaam, & Abdullah, 2015; Ferentinos, Katsoulas, Tzounis, Bartzanas, & Kittas, 2017; Jawad, Nordin, Gharghan, Jawad, & Ismail, 2017; Ojha, Misra, & Raghuwanshi, 2015), robotics (Ball et al., 2015) and Unmanned Aerial Vehicles (UAVs) or aerial drones (Mukherjee, Misra, & Raghuwanshi, 2019; Murugan et al., 2017). Moreover, Variable Rate Technologies (VRT) were used to optimize farming activities such as irrigation and seeding (Roberts et al., 2004).

Controlled Environment Agriculture (CEA) is an application of Precision Agriculture (PA). PA is about using technologies to enable farmers to electronically monitor, analyze and control soil, moisture and crop states. PA aims at optimizing the crop, taking into consideration factors such as weather, sun exposure, soil conditions and humidity. Almost two decades ago, Zhang et al. (Zhang, Wang, & Wang, 2002) provided a worldwide

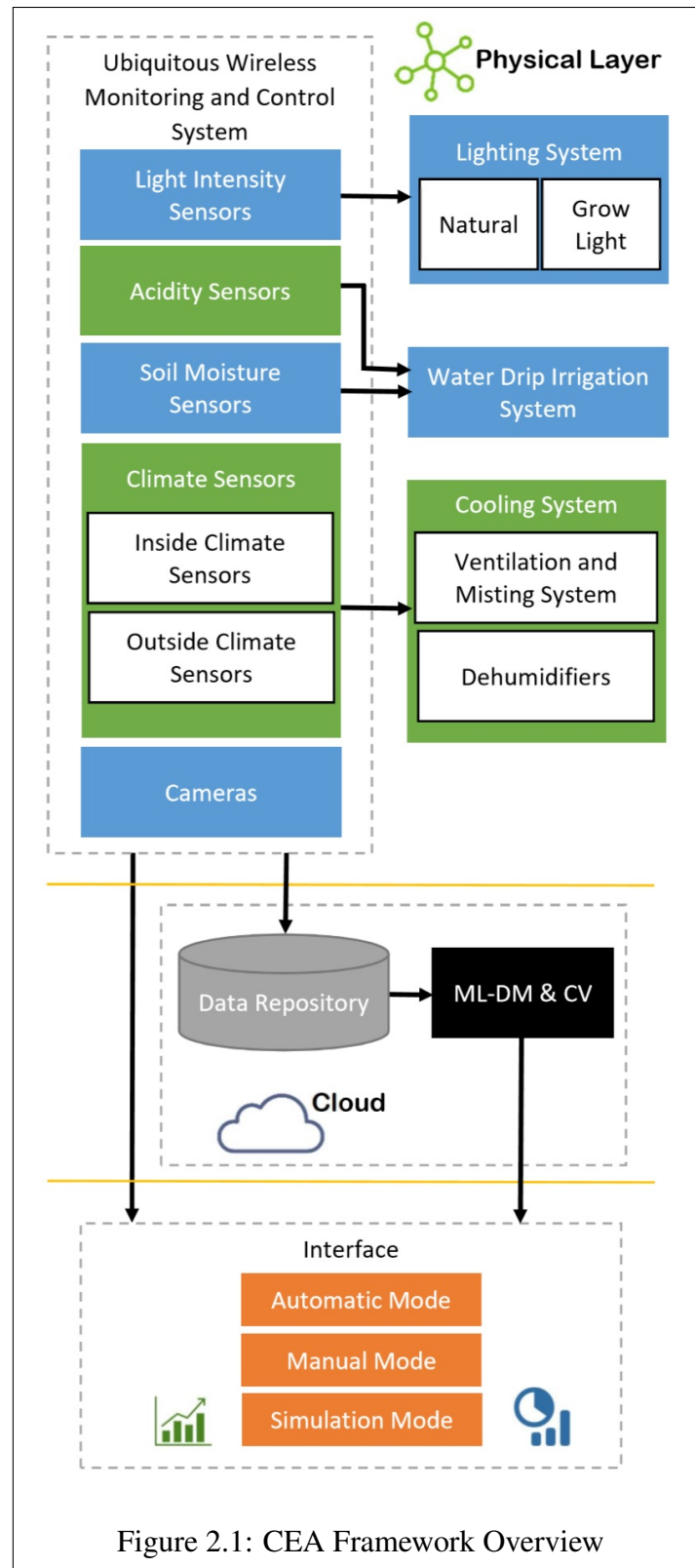


Figure 2.1: CEA Framework Overview

overview of the applications of PA technologies. Several studies about PA have been done in the Kingdom of Saudi Arabia, including (Al-Gaadi et al., 2016; Tola, Al-Gaadi, & Madugundu, 2019; Alameen, Al-Gaadi, & Tola, 2019; Patil & Al-Gaadi, 2012). What is common about these studies is the focus of field farming and the exclusion of greenhouse or indoor farming.

Modern greenhouses operate as systems. This is why a greenhouse is also referred to as a CEA system or a Controlled Environment Plant Production System (CEPPS). Shamshiri et al. (R Shamshiri et al., 2018) provided a review of the improvements in CEA technologies with a highlight on urban agriculture and a variety of its derivatives, including plant factories, rooftop farming and vertical farming. This study is focusing on soil-based greenhouse farming. Other varieties of CEA systems, such as soilless farming and vertical farming, are excluded and considered out of the scope of this study.

Fig. 2.1 represents what, generally, a CEA system consists of. A CEA system is a ubiquitous wireless monitoring and control system that interacts with several systems in the greenhouse. Moreover, the system consists of multiple sensors installed inside and outside the greenhouse. The coming subsections discuss the details of each subsystem within the greenhouse:

Lighting System. Since the GCC countries lie in the global sunbelt, natural sunlight is the main source of light in our greenhouse CEA system. Sunlight exposure intensity could be controlled by installing a shutter that opens or closes in response to light intensity sensor reading. However, grow lights can be used to enhance the growth of some plants when needed. Many farmers, in the region, use grow lights to boost the rate of plant growth. During the day time, plants receive a sufficient amount of sunlight. To accelerate the growth of the plants at night, grow lights can be used by the farmers. Moreover, farmers use different colors of light to improve crop quality. Various colors, or wavelengths, of light play different

roles for plant growth. The promotion of plant growth using a variety of artificial lights has been the focus of significant research and experimentation, as mentioned in (Maxik et al., 2015). For instance, the use of blue lights can increase the growth rate of plants more than other colors, hence it is preferable for farms nurseries to grow seedlings. Light intensity Sensors are used to measure the intensity of light exposure inside the greenhouse. Had the values given by the sensors reach a specific threshold, a horizontal shutter will be used to shade the plants inside the greenhouse. This mechanism is only needed because of the arid and extremely hot weather of the region.

Water Drip Irrigation System. To use water efficiently and effectively a water drip irrigation system is used. The irrigation system allows the water to target plant roots. Two water tanks are used; one underground tank to store water from multiple sources, and another overhead tank which is connected to a network of irrigation pipes. A pump is also used to pump water from the underground tank to the overhead tank. Water, whatever its source is, is not used directly to irrigate the plants. Instead, it is stored in the underground tank. A sensor is used in each of the two tanks to measure the level of water. A microprocessor switch is used to switch off the pump once the overhead tank is filled and another microprocessor switch is used to switch off the valve feeding the underground water once the underground water is filled. GCC countries are the world's poorest countries in natural freshwater resources (Darwish, Abdulrahim, Mohammed, & Mohtar, 2016). Despite the freshwater shortage, people in the GCC countries have full access to clean drinking water and water for sanitation purposes. The majority of domestic water supply in the GCC countries rely mainly on costly desalinated seawater which is used either directly or blended with groundwater. To help improve the sustainability of water resources for the greenhouse, rainfall harvesting could be used to collect water during the rainfall seasons. The use of two tanks allows the storing of the harvested water. The overhead

tank is connected to a network of drip irrigation pipes and can be controlled using a valve. Multiple sensors are used to measure the soil moisture and soil acidity. Depending upon soil moisture and acidity values given by the sensors, the valves of the irrigation water pipe will open or close. The valves will automatically open if the soil moisture is below a specific threshold and close when moisture has reached a specific threshold. Moreover, additional acidity sensors are used to measure the acidity of the water in the tanks to assure the quality of the water and its suitability for agricultural use.

Ventilation and Cooling Systems. The climate control of the greenhouse is done using two systems: ventilation and misting system and dehumidifying system. The ventilation and misting system consists of multiple fans and water pumps. Each fan is associated with a water pump. The fan can operate separately, without the water pump, if the purpose is to increase the level of ventilation of the greenhouse. However, if misting is the target, both the fans and associated water pumps will work together. On the other hand, the dehumidifiers are, most probably, needed in the very humid summers of Bahrain. Temperature and humidity are monitored using multiple sensors. Two sensors that sense temperature and humidity, respectively, are installed inside the greenhouse. Similarly, two more sensors for temperature and humidity are installed outside the greenhouse. If the values reach a specific threshold the climate control systems will be switched on or off automatically.

Cameras. Multiple cameras will be installed inside the greenhouse using a rail-mount system to capture close up images of the plants and crops at different times and locations in the greenhouse.

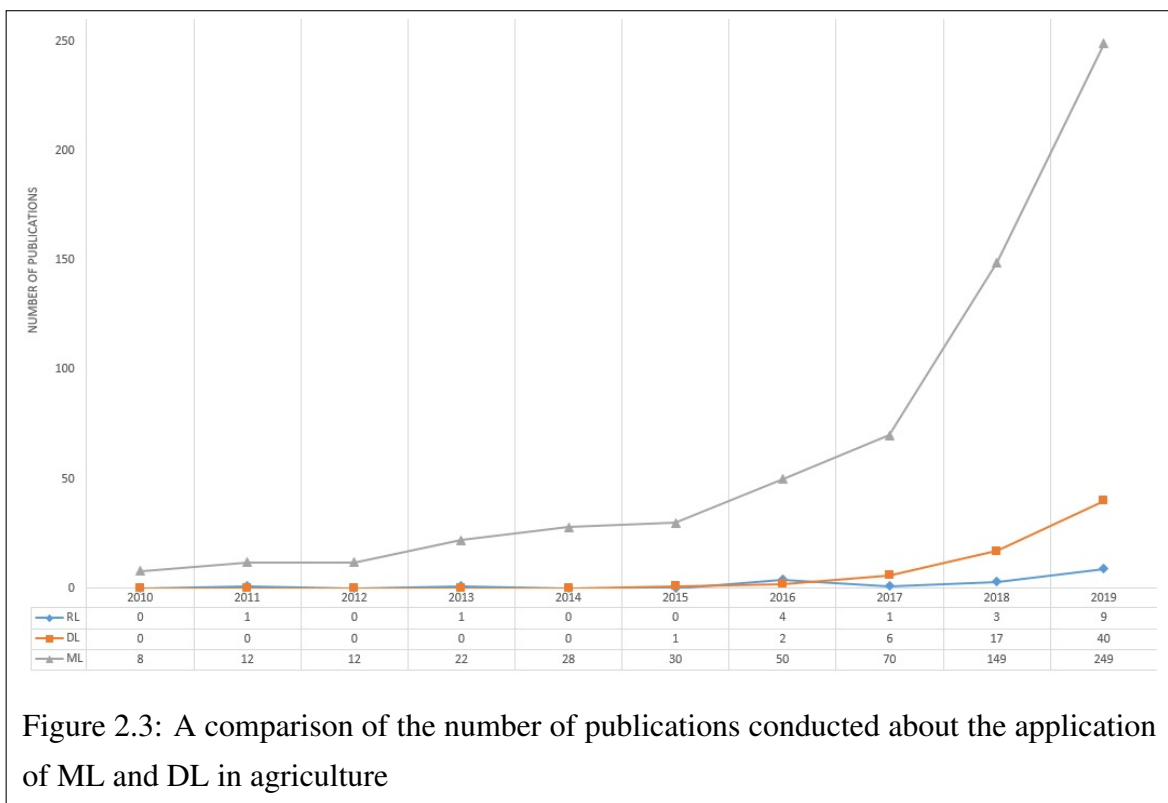
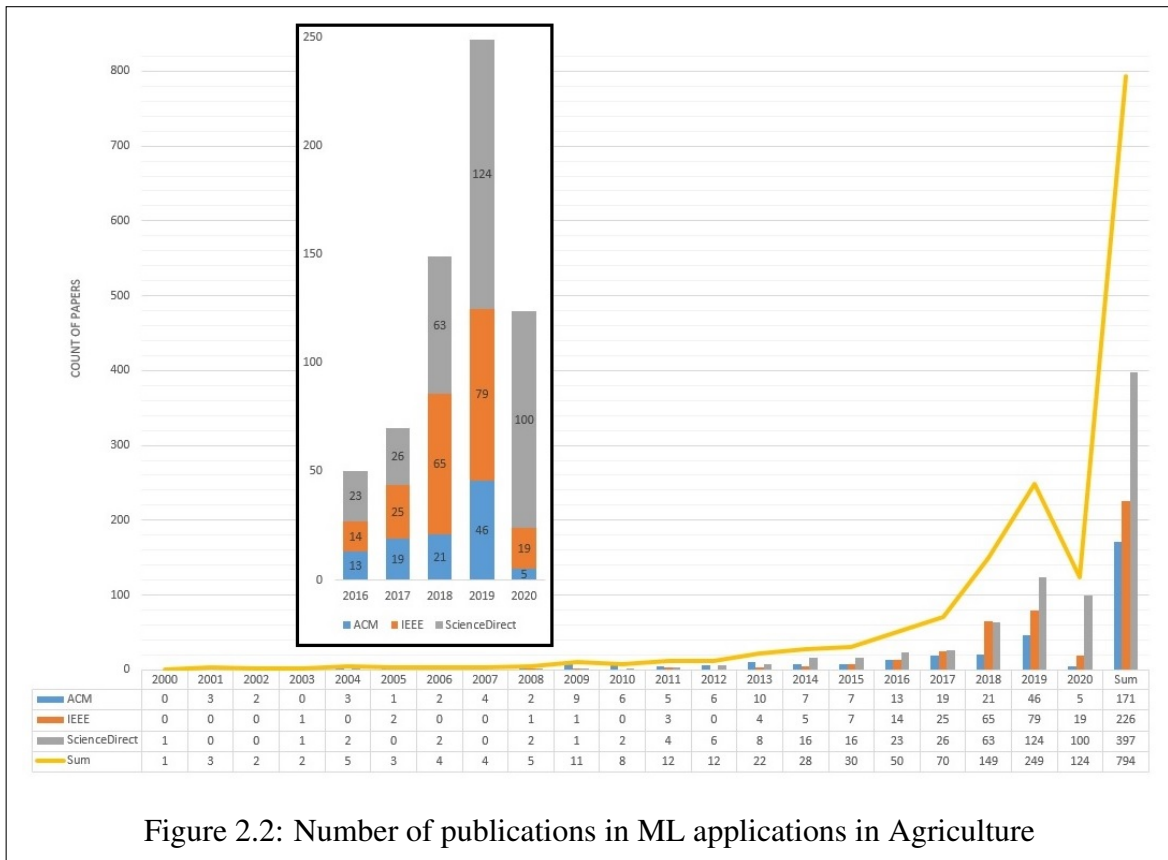


Table 2.1: Secondary research about ML applications in agriculture in the last 5 years

	<i>Ref.</i>	<i>Country</i>	<i>Type</i>	<i>DL</i>	<i>IoT</i>	<i>CV</i>	<i>CC</i>	<i>Application</i>
1	(Tripathi & Maktedar, 2016)	India	Survey	×	×	✓	×	Agricultural products disease detection and classification
2	(Shinde & Kulkarni, 2017)	India	Survey	×	✓	×	×	Crop disease prediction
3	(Kamilaris & Prenafeta-Boldú, 2018)	Spain	Survey	✓	×	✓	×	Improving agriculture using DL
4	(Jha, Doshi, Patel, & Shah, 2019)	India	Survey	✓	✓	✓	✓	Automating agriculture
5	(Batra & Gandhi, 2019)	India	Survey	×	×	×	×	Predictive modelling in protected cultivation
6	(Hungilo, Emmanuel, & Emanuel, 2019)	Indonesia	Survey	✓	×	✓	×	Plant disease detection and classification
7	(J. G. A. Barbedo, 2019)	Brazil	Survey	✓	×	✓	×	Plants nutrition deficiencies detection
8	(Weiss, Jacob, & Duveiller, 2020)	France, Italy	Meta review	✓	✓	✓	✓	Remote sensing for agricultural applications
9	(Sharma, Kamble, Gunasekaran, Kumar, & Kumar, 2020)	India	SLR	✓	×	✓	×	Improving agriculture supply chain using ML
10	(Ngugi, Abelowahab, & Abo-Zahhad, 2020)	Egypt	Survey	✓	×	✓	×	Leaf pest and disease recognition

2.3 Methodology

To get an initial idea about the current state of the research conducted in the field of ML applications in agriculture, a bibliographic analysis in the domain under study is conducted. The first step is to conduct a keyword-based search for conference papers and journal articles in three electronic databases: ACM digital Library, IEEE Xplore digital library, and ScienceDirect digital library. We used the following search commands distinctly:

”machine learning”.(agriculture+farming)

”deep learning”.(agriculture+farming)

where ‘.’ corresponds to the Boolean *and* operator, and ‘+’ corresponds to the Boolean *or* operator.

In the three libraries, a total of 794 papers about ML application in agriculture were published in the past twenty years. Figure 2.2 illustrates the steady growth of the number of publications during the years (2016 - 2020). Furthermore, Figure 2.3 gives a comparison of the number of publications conducted about the application of ML and DL, in agriculture the years (2000 - 2020).

The second step is to narrow the results to a shorter list of the most relevant papers. To do so we filter the results to papers published in the last five years only. Then we examined the shortlist particular problem they focused on and we ended having a list of 39 publications that we consider the most relevant to our study. And the third step is to analyze and compare the surveyed works based on the following: (1) the agricultural problem, (2) the learning method employed, (3) the data source used, and (4) the performance Evaluation methods.

2.3.1 Related Surveys

Table 2.1, summarizes ten secondary research publications that cover the applications of ML in agriculture in the years (2016 - 2020). Eight out of the ten publications were regular Survey efforts, one is a meta-review, and one is a Systematic Literature Review (SLR).

Tripathi and Maktedar (Tripathi & Maktedar, 2016) focused on ML and Computer Vision (CV) to study the classification and detection of plant diseases so that the agricultural products can be improved by controlling those diseases. This research provided a survey of seven research efforts that used several ML techniques, such as K-Means, Artificial Neural Networks (ANN), and Support Vector Machine (SVM), for detecting and classifying agricultural diseases. Most of the techniques were based on CV and image processing approaches. SVM was considered the best in predicting and classifying agricultural diseases.

Similarly, Shinde and Kulkarni (Shinde & Kulkarni, 2017) focused on crop diseases. However, they focused not only on classifying and detecting crop diseases but also on predicting and forecasting diseases. This research provided a survey of six research efforts that used ML techniques and the Internet of Things (IoT) technology to forecast crop diseases.

Batra and Gandhi (Batra & Gandhi, 2019) focused on predictive modeling so that decision making can be improved in protected agriculture. They highlight that in protected cultivation the micro-environment provided to the plants can help in controlling the negative effects of any factors. When predictive modeling is used in agriculture, it can help in proper water management and energy utilization. This research focused on two main ML algorithms, namely ANN and Bayesian Network (BN). ANN was the most common method used in predictions in greenhouses. Most of the surveyed works focused on the temperature control of the greenhouse and optimization of temperature maintenance costs.

2.4 Machine Learning in Agriculture

Table 2.2 shows the six surveyed publications that utilized conventional ML in different agricultural applications. The first three papers theoretically proposed an architectural design of systems to monitor, control, and/or automate farm activities. The other three papers provided evidence of experiments with some datasets to predict some features related to farm activities or detect and classify some plant diseases.

Several studies suggested utilizing several information technologies to fully or partially automate farm activities. Joshi et al. (Joshi et al., 2017), for instance, suggested the use of a bot to provide help and assistance in small scale farming. The bot can be used by the farmer who is planning to plant something in the backyard or in the garden. With the use of various machines, the bot can take over the process of farming from planting seeds to watering and weeding out the other plants in the field which are not required. Using sensors, the bot can sense the qualities that the soil has and it can also understand the weather of the area. This research utilized Bayesian Networks to automate farm activities. In a similar context, Kaburuan et al. (Kaburuan et al., 2019) highlighted the problems that agriculture faces in Indonesia. Climate change is one important issue that creates extreme weather conditions that can damage crops. The effect of the same is felt of the plant quantities and quality. The use of a technology-based monitoring system can improve the situation. This research presented an indoor micro-climate which has an electronic sensor to monitor the performance of horticulture. Sensor reading is collected regularly into a database that is linked with the data of meteorological agency. This research suggested the future of agriculture will depend more on ML. On the other hand, Vadivel et al. (Vadivel et al., 2019) focused on the monitoring and control of hydroponics. Various sensors are applied in this case using various algorithms of ML such as Simple Linear Regression to monitor and control hydroponics.

Table 2.2: Conventional ML applications in agriculture

	<i>Ref.</i>	<i>Year</i>	<i>Country</i>	<i>Type</i> ^a	<i>IoT</i>	<i>CV</i>	<i>CC</i>	<i>Application</i>
1	(Joshi et al., 2017)	2017	India	D	✓	✓	✓	Farm activities automation
2	(Kaburuan, Jayadi, et al., 2019)	2019	Indonesia	D	✓	×	×	Monitoring the performance of indoor micro-climate horticulture
3	(Vadivel et al., 2019)	2019	India	D	✓	×	✓	Monitoring and controlling hydroponics
4	(Helfer, Barbosa, Costa, Martini, & Santos, 2019)	2019	Brazil	B	✓	×	×	Productivity and soil fertility prediction
5	(Gertphol, Chulaka, & Changmai, 2018)	2018	Thailand	E	✓	×	×	Hydroponic farm lettuce quality prediction
6	(Hossain, Hossain, & Rahaman, 2019)	2019	Bangladesh	E	×	✓	×	Plant leaf disease detection and classification

^aD: Architectural Design, E: Experimental Research, B: Both

The focus of this research was the use of ML to suggest the optimal use of chemicals. Live data was harvested from the proposed system. The system is semi-automated so part of the data was entered manually. The data storage from this hydroponics system would be stored remotely which would help the consumers in knowing if the product is organic or not.

Conventional ML techniques and other computational technologies, such as image processing, have been used in literature to experiment with datasets and predict or detect some features related to the quality of the agricultural activities and products. Helfer et al. (Helfer et al., 2019), for instance, discussed the use of ubiquitous computing in agriculture. They suggested that highly sophisticated sensors can be used to study the soil properties and to predict productivity and soil fertility. This research focused on wheat farms and studied different climatic events that had an impact on the overall production of the wheat crop between 2001 and 2015. This research used the Partial Least Squares Regression. Similarly, Gertphol et al. (Gertphol et al., 2018) discussed the importance of yield forecasting in agriculture. However, this research is focused on Thailand agriculture where hydroponic lettuce farms were studied with excessive use of IoT. Several models were created to predict lettuce quality including linear regression, Support Vector Regression (SVR), Multiple Linear Regression (MLR), and ANN. Additionally, Hossain et al. (Hossain et al., 2019) focused on plant leaf diseases and the use of ML and CV to mitigate this threat. KNN classifier is proposed in this paper for disease detection and classification. The dataset used to train the model contained 237 images of leaf disease collected from 2 larger datasets. The images were labeled with five leaf-disease classes. The experiment proved successful in almost 96% of cases.

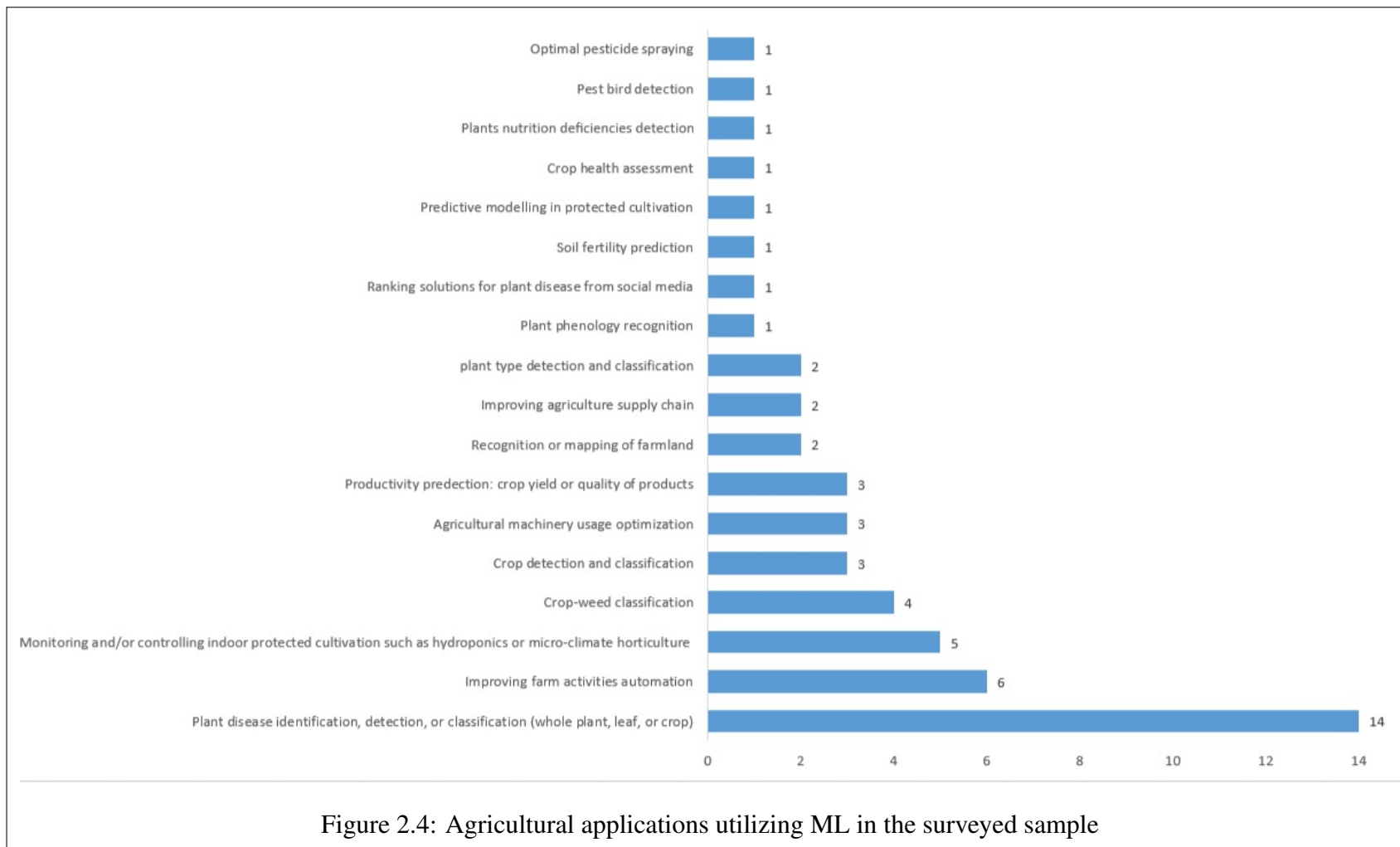


Figure 2.4 illustrates the variety of agricultural problems that research efforts in the surveyed sample addressed. The most common problem discussed in research is related to disease and pest identification and classification. However, there are agricultural problems that are scarcely tackled by the research community although these problems can be solved using ML techniques.

2.5 Deep Learning in Agriculture

DL uses deep neural networks of supervised and/or unsupervised learning algorithms to model high-level abstractions in data. It learns hierarchical representations in deep architectures for different tasks such as classification. DL models contain multiple layers of Artificial Neural Networks (ANN), such as convolutional layers. The input data, which is usually raw data, is fed into a network of multiple layers. The output of each layer is fed as an input to the next layer of the network. The output of the final layer can be used for classification. Each layer applies a nonlinear transformation on its input in order to extract underlying explanatory factors. DL algorithms allow the automation of feature extraction from the data.

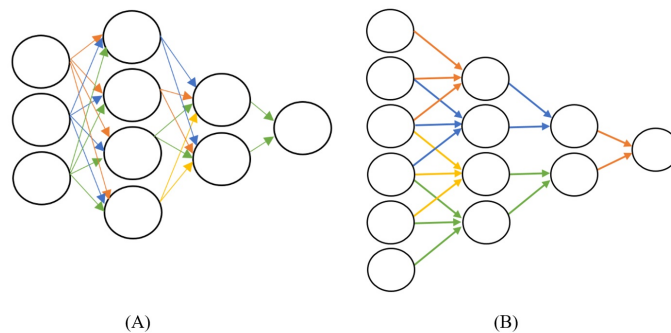


Figure 2.5: A generic architecture of traditional ANN and CNN

Convolutional Neural Networks (CNN) is a form of deep neural networks that are commonly used in different learning tasks, such as image

recognition and image classifications. Like an ANN, a CNN consists of multiple layers. However, a traditional ANN usually consists of a fully connected network, that is, each neuron in one layer is connected to all neurons in the next layer, as demonstrated in Fig 2.5-A. A CNN, on the other hand, takes advantage of the hierarchical pattern in data and assemble more complex patterns using smaller and simpler patterns. Fig. 2.5-B shows how in CNN, instead of a neuron being connected to all neurons of the previous layer they are connected to only neurons close to it.

Kamilaris and Prenafeta-Boldú (Kamilaris & Prenafeta-Boldú, 2018) performed a survey of 40 research efforts that employed DL techniques to various agricultural and food production challenges. This review paper discussed many DL techniques, such as CNN, Long Short Term Memory (LSTM), and Gated Recurrent Units (GRU). The research discussed the applications of CV in agriculture and popular techniques used for that. The agricultural problems discussed in this study include leaf classification, leaf disease detection, plant disease detection, land cover classification, crop type classification, plant recognition, plant phenology recognition, segmentation of root and soil, crop yield estimation, fruit counting, obstacle detection, weed identification, crop/weed detection and classification, prediction of soil moisture content, animal research, and weather prediction.

Ngugi et al. (Ngugi et al., 2020) provided a review of recent studies carried out in the area of automated leaf pest and disease recognition. They discussed the use of image processing techniques and DL methods to address this task. The scope of the study, though, is limited to image processing techniques employed in pest and disease recognition using visible light (RGB) images. This study compared 17 works on leaf disease recognition using handcrafted feature extraction and shallow classifiers. Moreover, this study summarized 34 published studies in crop pest and disease recognition using DL algorithms.

Hungilo et al. (Hungilo et al., 2019) argued that Tanzania can improve its agricultural production efficiency by early detection and classification of diseases. This research focused on the use of image processing and ML models for disease classification. The purpose here is to ensure that plant disease identification can be done on time and production can be improved. This research provided a survey of fifteen research efforts among which six research efforts used CNN and nine research efforts used conventional methods such as the K-Mean clustering algorithm.

Jha et al. (Jha et al., 2019) provided a survey of ten research efforts in which they discussed the importance of agricultural automation to all countries. In this work, five main agricultural problems were addressed, namely plant disease infestation, lack of storage management, pesticide control, weed management, and lack of irrigation and drainage facilities. This work focused on integrating the technologies of conventional ML, DL, CV, IoT, AI, and Cloud Computing (CC) to automate agricultural activities and improve agricultural production and efficiency. Many DL and conventional ML algorithms were discussed in this work such as CNN, ANN, K-Nearest Neighbors (KNN), Fuzzy Logic, Neuro-Fuzzy Logic, and Expert Systems.

Weiss et al. (Weiss et al., 2020) discussed that remote sensing can be used in the evolution of agriculture. According to this research, there are various variables in agronomics that can be studied through remote sensing. It also suggests several applications of remote sensing in agriculture such as phenotyping and breeding, monitoring of agricultural land usage, crop yield forecasting, optimization of short-term production, and the provision of ecosystem services related to soil or water resources as well as to animal or plant biodiversity. This research effort is a meta-review that focused mainly on remote sensing applications in agriculture. Moreover, it discussed the integration of ML, IoT, CV, and CC technologies to improve smart farming.

Sharma et al. (Sharma et al., 2020) conducted an SLR on using ML to improve decision making in agriculture. Decision making can be improved due to the availability of data on the basis of which agricultural decisions can be taken. This SLR studied 93 papers that show that ML can help in the development of sustainable supply chains in agriculture. Many ML algorithms were addressed in this study such as BN, SVM, ANN, Decision Trees (DT), ensemble learning, regression analysis, clustering, genetic algorithm, and many deep learning algorithms.

Table 2.3 put together the 25 surveyed research papers that utilized DL techniques in different agricultural applications. The general notice is that in the majority of sampled publications DL, CV, and data processing are used to analyze images datasets related to different agricultural activities and products. More than a third of the surveyed sample of publication is utilizing DL to detect or classify some sort of disease or pest affecting plants or crops.

Wan et al. (Wan, Lu, Qi, & Chen, 2020) stated that the economic value of food loss with reference to plant disease is in billions in China only. If the food loss in other parts of the world is considered the economic loss is much bigger. They argued that food loss in the world should be controlled and that the use of technology is highly recommended for pest control and disease control. By utilizing the right detection methods for plant diseases, the production of agriculture can be improved.

Another research on a related topic is done by Saravan and Perepu. This research presents social media as a source of information for the farmers (Saravanan & Perepu, 2018). The texts and images from the fields can be shared on social media to generate in-depth knowledge about the disease. In this study, CNN was employed to identify plant disease from images, and rank regression was used to rank solutions from social media.

Table 2.3: DL applications in agriculture

	<i>Ref.</i>	<i>Country</i>	<i>Type</i> ¹	<i>IoT</i>	<i>CV</i>	<i>CC</i>	<i>Application</i>
1	(Carvalho et al., 2019)	Ireland	D	✓	✓	✓	Edge solution for farming
2	(Mehra, Saxena, Sankaranarayanan, Tom, & Veeramankandan, 2018)	India	B	✓	×	✓	Controlling hydroponics system
3	(Saravanan & Perepu, 2018)	India	E	×	✓	×	Plant disease identification and social media solutions ranking
4	(Wan et al., 2020)	China	E	×	✓	×	Plant disease classification
5	(Jain, Vardhan, Nishanth, & Shylaja, 2017)	India	E	×	✓	✓	Plant disease classification
6	(Yalcin, 2017)	Turkey	E	×	✓	×	Plant phenology recognition
7	(Xu, Wei, Ye, & Wang, 2019)	China	E	×	✓	×	Recognition of <i>Zanthoxylum Armatum</i> rust
8	(Alfarisy, Chen, & Guo, 2018)	China	E	×	✓	×	Paddy pests and diseases recognition

¹D: Architectural Design, E: Experimental Research, B: Both

9	(Shuangpeng, Tao, & Hong, 2019)	China	E	×	✓	×	Farmland recognition
10	(Xin & Adler, 2019)	USA	E	×	✓	✓	Identifying Miscanthus
11	(A. X. Wang, Tran, Desai, Lobell, & Ermon, 2018)	USA	E	×	✓	×	Crop yield prediction with remote sensing data
12	(Bhatt, Sarangi, & Pappula, 2017)	India	E	×	✓	×	Crop health assessment
13	(Miao, Zheng, & Tao, 2019)	China	E	×	✓	×	Crop weed identification
14	(Zhou, Li, & Shao, 2018)	China	E	×	✓	×	Crop classification
15	(S. Lee, Lee, Jeon, & Smith, 2019)	Korea	E	×	✓	×	Bird identifying from other moving objects
16	(Ale, Sheta, Li, Wang, & Zhang, 2019)	USA, China	E	×	✓	×	Plant disease detection
17	(Fawakherji, Youssef, Bloisi, Pretto, & Nardi, 2019)	Italy	E	×	✓	×	Crop and weeds classification
18	(Suh, Ijsselmuiden, Hofstee, & van Henten, 2018)	Netherlands	E	×	✓	×	Sugar beet and volunteer potato classification

19	(Habaragamuwa et al., 2018)	Japan	E	×	✓	×	Greenhouse strawberries detection and classification
20	(Coulibaly, Kamsu-Foguem, Kamissoko, & Traore, 2019)	France	E	×	✓	×	Mildew disease identification in pearl millet crop
21	(Santos, de Souza, dos Santos, & Avila, 2020)	Brazil	E	×	✓	×	Grape detection and segmentation
22	(Sidike et al., 2019)	USA	E	×	✓	×	Mapping heterogeneous agricultural landscape
23	(Kim, Lee, & Kim, 2020)	Korea	E	×	✓	×	Onion downy mildew disease detection
24	(dos Santos Ferreira, Freitas, da Silva, Pistori, & Folhes, 2019)	Brazil	E	×	✓	×	Weed discrimination
25	(Picon, Seitz, et al., 2019)	Spain, Germany	E	×	✓	×	Multi-crop plant disease classification

Similarly, Jain et al. (Jain et al., 2017) suggested that the utilization of ML methods and the provision of a cloud-based system can help farmers in understanding if their crop is healthy or not. The study argued that using CNN, not only plant diseases can be studied but the use of pesticides and chemicals can also be controlled. In this research, a dataset of manually obtained images of firecracker and pomegranate plants from southern India was used. The images in the dataset were classified as diseased or healthy. Two models were used, CNN and SVM, in the plant classification task.

Likewise, according to Alfarisy et al., (Alfarisy et al., 2018) Paddy production in Indonesia is harmed greatly by pests and diseases. The automatic identification of pests and diseases is still a challenge in paddy production in Indonesia. By applying DL, the task of paddy pest and disease recognition can be resolved. For this purpose, this study used a dataset of images collected from the internet. The dataset was collected using a query of pests and diseases name in different languages that represent the majority of paddy producers in the world. The dataset, then, was annotated by agriculture experts into 13 classes. CaffeNet which is a CNN model was used for the recognition task.

In a similar manner, in the research by Xu et al. (Xu et al., 2019) the focus was on the recognition of one disease that affects one plant, which is the *Zanthoxylum Armatum* rust. A dataset consisting of 22,937 images was used in this research. The dataset had a total of 19 kinds of leaf diseases and it was labeled as healthy or rust. In a similar way, Bhatt et al. (Bhatt et al., 2017) stressed that the timely diagnosis of plants nutrients deficiencies and plant disease helps improve yield. Low cost, reliable and quick diagnoses are needed. This research developed a participatory application for sensing using smartphones. Farmers can scout their fields and assess crop health in this way. CNN is used to classify plant images. Also, transfer learning used to calculate the features of the images for classification.

Picon et al. (Picon, Seitz, et al., 2019) used also used CNN for multi-crop plant disease classification. Cell phones acquired images in the field were taken in real field wild conditions. Coulibaly et al. (Coulibaly et al., 2019) also stressed that classification is important in agriculture and DL can be effective in this case only if transfer learning is applied. This research aimed to identify mildew disease in pearl millet crop. CNN with transfer learning was used to achieve this task.

In like manner, Kim et al. (Kim et al., 2020) argued that image based-monitoring system for agricultural fields can monitor crops automatically. Onion fields were monitored in their experiment. CNN model was utilized to detect onion downy mildew disease. Images used as input to the model were captured onion images using the filed monitoring system. Finally, as per Ale et al. (Ale et al., 2019) in smart agriculture, DL is a smart approach. In this research, a Densely Connected Convolutional Networks (DenseNet) was proposed for transfer learning.

Another factor that can cause lower plant productivity in farming is related to poor weed control. That is why many efforts were dedicated to utilizing DL and other computational technology for weed identification and detection. As discussed by Miao et al. (Miao et al., 2019) to handle weed effectively, it is important to have a clear distinction between crops and weeds. Based on this background, Miao et al. (Miao et al., 2019) introduced a CNN to classify weed and crop by using images from an open dataset. Another research that focused on weed control is the one conducted by Suh et al. (Suh et al., 2018). They stressed that weed control can have a direct impact on the yield. The focus of there experiment was to classify sugar beets and volunteer potatoes. Volunteer potatoes can be controlled using smart bots. CNN is used in this study to classify sugar beet and volunteer potato. As per do Santos Ferreira et al. (dos Santos Ferreira et al., 2019) the use of UAVs and unsupervised DL algorithms can prove to be cost-effective. In

this study, image recognition with CNN was used for weed discrimination. Similarly, Fawakherji et al. (Fawakherji et al., 2019) focused on the use of robotics in agriculture for taking care of weeds. For this purpose, the CNN technique is used to classify weeds and crops. Another issue that concern farmers is the effective control of bird pest in agriculture. Lee et al. (S. Lee et al., 2019) focused on detecting pest birds in the fields of agriculture. It is considered to be difficult to identify birds in the dynamic background. The objects that mostly confuse bird detection in agriculture environments are flying things, trees, and clouds.

Remote sensing and satellite imagery are widely used in the field of agriculture. As per Zhou et al. (Zhou et al., 2018) CNN can help in classifying different land covers using Remote Sensing Images (RSI). This research focused on the comparison of the spatial distribution of crops in a county in China using Sentinel-2A RSI. The satellite imagery contained spectral information and temporal information. In a similar manner, Sidike et al. (Sidike et al., 2019) proposed a model based on CNN for mapping heterogeneous agricultural landscape through remote sensing. Also, as discussed by Shuangpeng et al. (Shuangpeng et al., 2019), for estimation of yields of grain, farmland mapping is vital. In this research, a Deep Feature Aggregation Net (DFANet) was used to recognize the farmlands. DFANet is a CNN architecture. Furthermore, Xin and Adler (Xin & Adler, 2019) discussed the role of grasslands in agriculture as well as in ecology. Similarly, phenology is important because it affects whether plants thrive or survive in their environments and it is important because our food supply depends on the timing of phenological events. Yalcin (Yalcin, 2017) focused on monitoring the phenology to improve precision in the field of agriculture. When phenological changes in plants can be identified quickly, the yield output and harvesting time can be predicted accurately. CNN used in this case to recognize plant phenology.

Another important application of DL in agriculture is the detection of farm crops. Santos et al. (Santos et al., 2020), for instance, used CNN for grape detection and segmentation. The same technology is used in segmentation and tracking too. Another related research is conducted by Habaragamuwa et al. (Habaragamuwa et al., 2018). They discussed that generic algorithms are needed in agriculture to detect and classify objects. In this experiment, DL was utilized to study the mature and immature strawberries by classifying them using CNN and greenhouse imagery.

Designing and implementing smart solutions to monitor and control farms and optimize farming activities is an important research topic. Carvalho et al. (Carvalho et al., 2019) proposed an edge solution for farming. They discussed that edge computing, IoT, and CC can help in making smart agriculture possible. And they argued that to create smart agricultural solutions the integration of ML algorithms is needed such as Deep neural networks (DNN), CNN, Recurrent neural network (RNN), and Genetic Algorithm (GA). Moreover, Mehra et al. (Mehra et al., 2018) stressed on the fact that modern farming techniques need to rely on the latest technologies such as aeroponic and hydroponic. This research highlights that the use of IoT and ANN and other ML algorithms helps in controlling hydroponics systems.

Crop yield prediction is important in developing countries so that agriculture can be developed sustainably. The current technology that is commonly used in this regard is expensive. Remote sensing and satellite imagery can provide a much cheaper option. Wang et al. (A. X. Wang et al., 2018) developed a model based on LSTM and regression to predict soybean yield in Argentina. For experimentation, annually updated satellite reflectance and temperature imagery were used to identify cropland from non-cropland.

2.6 Deep Clustering Algorithms

Deep clustering algorithms are a class of unsupervised learning algorithms that combine deep neural networks with clustering techniques. The goal of these algorithms is to find meaningful latent representations of the input data and arrange them into discrete groups. Deep clustering methods are designed to learn a representation of the data that encompasses both local and global structures. This is accomplished by training a deep neural network with multiple layers on the input data to extract hierarchical characteristics. Each layer of the network learns progressively more abstract representations of the input, allowing complex patterns and structures to be discovered.

Deep clustering algorithm training generally consists of two major steps: representation learning and clustering. The deep neural network is trained to map the input data to a lower-dimensional latent space during the representation learning stage. This latent space is intended to capture the most essential characteristics of the data, allowing for improved clustering performance. To learn these latent representations, several algorithms such as autoencoders or variational autoencoders are frequently used. Following the completion of the representation learning process, the clustering step is done to assign each data point to a distinct cluster. Traditional clustering algorithms, such as k-means or Gaussian mixture models, are frequently utilized, with the learned latent representations serving as input features. The deep clustering technique seeks to improve both the representation learning and clustering steps jointly, often using an iterative process to refine the clustering results.

Deep clustering techniques have the benefit of automatically learning meaningful patterns of data without the need for manual feature engineering. These algorithms can identify complicated patterns that are not immediately evident from raw input data. This makes them especially effective in dealing with high-dimensional and complicated data, like image analysis.

In this section, we are providing a theoretical background of deep clustering algorithms that integrates deep learning and clustering in an end-to-end manner (J. Yang et al., 2016; Z. Jiang, Zheng, Tan, Tang, & Zhou, 2016; B. Yang et al., 2017; Ghasedi Dizaji, Herandi, Deng, Cai, & Huang, 2017; X. Guo et al., 2017; X. Guo, Zhu, Liu, & Yin, 2018; Sarfraz, Sharma, & Stiefelhausen, 2019; Mrabah, Bouguessa, & Ksantini, 2020; Mrabah, Khan, Ksantini, & Lachiri, 2020). A detailed explanation of the models that are used in this study is provided in the next chapter.

Joint Unsupervised LEarning (JULE) is a deep recurrent framework for jointly learning unsupervised representations using CNN and clustering images using agglomerative clustering (J. Yang et al., 2016). JULE’s key contribution is the recurrent process that allows clusters to be merged in numerous time steps. Training a recurrent neural network with the number of timesteps equal to the number of data points, on the other hand, is not computationally efficient.

Variational Deep Embedding (VaDE) is a generative deep clustering approach that leverages Variational Autoencoder (VAE) (Z. Jiang et al., 2016). It enables the combination of clustering and data generation. The VaDE methodology entails encoding the initial data distribution in the embedded space as a Gaussian Mixture distribution. The Gaussian Mixture distribution is sampled using the reparameterization approach to facilitate backpropagation. VaDE employs variational inference. As a result, the information loss caused by the mean-field approximation might degrade the quality of the latent space, resulting in poor clustering performance. VaDE introduces a variational lower bound objective that combines the reconstruction loss, clustering objective, and a regularization term.

Deep Clustering Network (DCN) (B. Yang et al., 2017) is an autoencoder clustering algorithm. DCN, features a joint optimization procedure. To begin, the autoencoder is pretrained to lower the dimensionality of the dataset using a reconstruction loss function. The ultimate objective of this method is to obtain a K-Means-friendly representation at the conclusion of the training procedure. A K-Means loss is used in conjunction with the vanilla reconstruction. This approach necessitates hard clustering assignments (as opposed to probabilistic soft clustering assignments). This would result in a discrete optimization process, which is incompatible with gradient descent's differential aspect.

Deep Embedded Regularized Clustering(DEPICT) (Ghasedi Dizaji et al., 2017) uses a convolutional autoencoder to learn embedded representations and clustering assignments. DEPICT provides an objective function based on relative cross-entropy (KL divergence). Furthermore, the DEPICT loss function includes a regularization term that allows you to directly impose the size of each cluster based on some prior information. This prevents scenarios in which the majority of the data points are allocated to a few clusters. However, DEPICT's proposed previous knowledge (cluster size) is unrealistic for pure unsupervised tasks. Furthermore, DEPICT ignored the clustering-reconstruction trade-off.

Deep Embedding for Clustering (DEC) (X. Guo et al., 2018), like DEPICT, provides an objective function based on relative cross-entropy (KL divergence). DEC uses an autoencoder to project data points in a lower-dimensional space. The decoder is then discarded, and the encoder is taught to enhance both the embedded representations and the clustering centers. Before beginning to cluster the data, DEC requires a pretraining step. Furthermore, after discarding the decoder, it lacks a means to avoid Feature

Randomness. As a result, a large number of random labels can result in the creation of random discriminative features. Furthermore, the loss function contribution of each center is normalized to prevent huge clusters from changing the embedded representations. As a result, DEC is more suited for balanced datasets.

Improved Deep Embedded Clustering (IDEC) (X. Guo et al., 2017) is an extension of DEC that addresses some of its limitations. However, IDEC, like DCN, lacks a way to reduce the clustering-reconstruction trade-off. It introduces a soft assignment strategy that allows data points to have multiple cluster assignments with varying degrees of membership. This helps to capture the uncertainty in the clustering process and provides a more flexible representation of the data. IDEC also incorporates a reconstruction loss in addition to the target distribution, which further improves the quality of the learned representation. According to Guo et al. (2017), removing reconstruction during the clustering step hinders the network’s ability to preserve local structures. However, we suggest that because of the intrinsic tension between clustering and reconstruction, retaining the reconstruction loss end-to-end is not as useful. More specifically, embedded clustering ignores non-discriminative patterns that do not help to data classification (for example, between-class similarities), whereas reconstruction is primarily concerned with retaining all information, discriminative or not. The preservation of nondiscriminatory traits from beginning to finish suggests a decline in total discriminative capacity.

First Integer Neighbor Clustering Hierarchy (FINCH) (Sarfraz et al., 2019) employs an autoencoder architecture to learn a low-dimensional representation of the input while retaining its significant characteristics. Finch then uses the learnt representation to use a clustering method, like k-means,

to group related data points together. Finch proposes to increase clustering performance by concurrently optimizing the reconstruction loss and clustering objective.

Adversarial Deep Embedded Clustering (ADEC) is an autoencoder-based clustering method (Mrabah, Bouguessa, & Ksantini, 2020). It is the first attempt to describe and address the trade-off between Feature Drift and Feature Randomness. ADEC, like DEC, minimizes the Kullback-Leibler (KL) divergence to a secondary target distribution. An adversarially restricted reconstruction also helps to regularize ADEC’s clustering goal. The conflict between the two objective functions is reduced by adversarial training. However, due to its instability, adversarial training remains a difficult process.

Dynamic Autoencode (DynAE) is a deep clustering technique that uses temporal dynamics to cluster data (Mrabah, Khan, et al., 2020). It employs an autoencoder design that considers the temporal connections between successive data points. DynAE tries to capture the underlying temporal patterns in the data by concurrently optimizing the reconstruction loss and a temporal consistency loss. This enables more accurate and robust grouping, especially in cases where data changes over time.

Deep clustering algorithms, in brief, employ deep neural networks to learn hierarchical representations of input data, which are subsequently used to perform clustering. These algorithms may identify significant patterns and structures in data by combining representation learning with clustering, making them a valuable tool in unsupervised learning tasks.

2.7 Deep Clustering for Plant Disease Detection

In recent years, the agricultural sector has witnessed a transformative shift in leveraging advanced technologies to address challenges related to crop health (Nguyen et al., 2021). One noteworthy area of innovation is the application of deep learning techniques in the domain of plant disease discovery (L. Li, Zhang, & Wang, 2021). Deep learning, a subset of machine learning, has demonstrated remarkable capabilities in handling complex tasks such as image analysis, classification, and pattern recognition (Y. Yang et al., 2018). This section aims to provide an overview of the current state of research on using deep learning for plant disease discovery, with a focus on deep clustering.

Barbedo (J. G. A. Barbedo, 2019) discussed that in recent times, digital images have been used to overcome agricultural challenges. ML has helped in identifying plant diseases and pests. Proximal images can provide more information including the situation of nutrients that are provided to the crops. Barbedo's research provided a survey of 46 research efforts that used any kind of sensor imagery such as visible range, multispectral, hyperspectral, and chlorophyll fluorescence, provided that images were captured at close range. According to Barbedo (J. G. Barbedo, 2022), in plant disease assessment, images, and deep learning, together, can be used in three main tasks: disease classification, detection, and severity estimation. A classification task involves determining which disease is responsible for a symptom or sign. Usually, the disease detection task is approached as a binary classification task that categorizes everything other than the disease of interest as a second class. After the classification or identification, the disease symptom severity estimation attempts to provide objective measurement of the extent of symptoms observed, for instance, the percentage of leaf area. To the three categories discussed by Barbedo (J. G. Barbedo, 2022), we add a fourth category: plant disease categorization using clustering techniques.

Table 2.4: Related works comparison

	Ref.	Plant Type	Dataset	Task	Deep Method
1	(K. Li, Lin, Liu, & Zhao, 2020)	Ginkgo	Own	Classification	VGG16 InceptionV3
2	(F. Jiang, Lu, Chen, Cai, & Li, 2020)	Rice	Own	Classification	CNN+SVM
3	(Bi et al., 2022)	Apple	Own	Classification	MobileNet InceptionV3 ResNet152
4	(Ahmad, Hamid, Yousaf, Shah, & Ahmad, 2020)	Tomato	Own	Classification	VGG-17 VGG-19 InceptionV3 ResNet
5	(Qiu et al., 2019)	Wheat	Own	Classification	Mask RCNN
6	(S. H. Lee, Goëau, Bonnet, & Joly, 2020)	Multiple	PV IPM-Bing	Classification	Mask RCNN
7	(Liang, Zhang, Zhang, & Cao, 2019)	Rice	Own	Detection	CNN
8	(Liu & Wang, 2020)	Tomato	Web crawler	Detection	MobileNetv2 YOLOv3
9	(Afzaal, Bhattarai, Pandeya, & Lee, 2021)	Strawberry	Own	Detection	Mask R-CNN
10	(Picon, Alvarez-Gila, et al., 2019)	Wheat	Preexisting Extended	Detection	Deep Residual Neural Network

11	(D. Li et al., 2020)	Rice	Own	Detection	RCNN VGG16 ResNet-50 ResNet-101 YOLOv3
12	(Yao et al., 2022)	Kiwifruit	Own	Detection	YOLOX
13	(D. Shah, Trivedi, Sheth, Shah, & Chauhan, 2022)	Multiple	PV	Detection	ResTS
14	(K. Li, Zhang, Li, Li, & Ma, 2022)	Cucumber	Own	Severity	Semantic segmentation DeepLab V3+
15	(C. Wang et al., 2021)	Cucumber	Own	Severity	DeepLabV3+ U-Net
16	(J. Gao et al., 2021)	Potato	Own	Severity	encoder-decoder architecture
17	(Xiang, Liang, Sun, Zhang, & Wang, 2021)	Multiple	PV Disease Severity	Severity	L-CSMS
18	(Esgario, Krohling, & Ventura, 2020)	Coffee	Own	Severity	AlexNet GoogLeNet ResNet50 VGG16
19	(Gokulnath & Usha Devi, 2020)	Maize	PV PDD	Categorization	DEPICT
20	This work	Multiple	PV	Categorization	DynAE

After reviewing the literature, we found that researchers seldom use clustering or deep clustering algorithms for categorizing plant diseases. Table 2.4 shows an overview of the related works that use RGB images and deep learning to tackle issues related to plant disease. Other than the work done by Gokulnath and Usha Devi (Gokulnath & Usha Devi, 2020), none of the presented experiments used deep clustering techniques to tackle plant disease assessment tasks. The limited research done in deep clustering for plant disease categorization motivates us to experiment with deep clustering techniques to deal with the problem of plant disease assessment.

Plant disease classification is a fundamental aspect of disease discovery, enabling farmers to identify and address specific threats to crop health promptly. (Asaari et al., 2018) stated that deep learning models, particularly CNNs, have proven highly effective in automating the classification process based on visual symptoms exhibited by plants. Researchers have developed models capable of accurately distinguishing between various diseases and even differentiating them from non-disease-related stress factors. The utilization of large datasets containing annotated images of diseased plants has played a crucial role in training these models, enabling them to generalize well across different crops and diseases. The success of deep learning in plant disease classification has opened avenues for the development of user-friendly mobile applications and tools that empower farmers with real-time information, facilitating timely interventions and minimizing crop losses. Plant disease classification plays a pivotal role in the field of disease discovery, offering farmers a crucial tool for the timely identification and mitigation of threats to crop health. As plants are susceptible to a myriad of diseases, ranging from fungal and bacterial infections to viral and nematode infestations, the ability to accurately categorize these diseases is essential. Deep learning models, CNNs, have proven to be adept at automating the classification process based on visual symptoms exhibited by plants. By

analyzing high-resolution images of diseased plants, these models can discern subtle patterns indicative of specific diseases, even differentiating them from non-disease-related stress factors. The implications of accurate plant disease classification are far-reaching, allowing farmers to implement targeted and effective control measures. Armed with this knowledge, farmers can respond promptly to emerging threats, reducing the risk of crop losses and ensuring the overall health and productivity of agricultural systems. The integration of deep learning in plant disease classification not only enhances the precision of identification but also contributes to the development of advanced technologies that empower farmers with real-time information, ultimately fostering sustainable and resilient agricultural practices.

We found that the plant disease classification task is the most explored in the literature. There is a high level of redundant research focusing on this task. A classification task usually utilizes CNNs (LeCun, Bengio, et al., 1995) classical architecture such as AlexNet (Krizhevsky, Sutskever, & Hinton, 2017), and Visual Geometry Group (VGG) (Simonyan & Zisserman, 2014) or a modern network architecture such as Inception (GoogLeNet) (Szegedy et al., 2015), and Residual Networks (ResNet) (He, Zhang, Ren, & Sun, 2016). For instance, Li et al. (K. Li et al., 2020) compared the application of VGG-16 and InceptionV3 to classify Ginkgo blight disease using a self-acquired dataset that contains images taken under both laboratory and field conditions. In a similar context, Jiang et al. (F. Jiang et al., 2020) experiment focused on classifying four rice leaves diseases using own dataset. Similarly, Bi et al. (Bi et al., 2022) used a self-acquired images dataset to classify apple leaf diseases. They used MobileNets (Howard et al., 2017) and compared the results with other CNN models (InceptionV3 and ResNet152). Ahmed et al. (Ahmad et al., 2020) utilize self-collected tomato leaves images from the lab and the field to classify tomato disease using several CNNs architectures, including VGG-17, VGG-19, InceptionV3,

and ResNet. Both the experiments of Qiu et al. (Qiu et al., 2019) and Lee et al. (S. H. Lee et al., 2020) utilizes a Mask R-CNN (He, Gkioxari, Dollár, & Girshick, 2017). Mask R-CNN is a Region-Based network. While Qiu et al. used a self-collected wheat images Lee et al. used images of multiple crop types collected from both PlantVillage and IPM and Bing datasets.

Plant disease detection involves the identification of early signs of infection, often imperceptible to the naked eye. Chowdhury et al. (Chowdhury et al., 2021) identified that deep learning techniques have demonstrated significant potential in automating this process by analyzing high-resolution images of plants and capturing subtle variations indicative of diseases. Guo et al. (Y. Guo et al., 2020) highlighted that one of the key advantages of deep learning in disease detection is its ability to learn hierarchical representations, allowing the models to discern intricate patterns that might be overlooked by traditional methods. Transfer learning, a technique where pre-trained models are fine-tuned for specific tasks, has emerged as a valuable approach in plant disease detection (Nagaraju & Chawla, 2020). Feng et al. (Feng et al., 2020) examined that by leveraging knowledge gained from large datasets, these models can adapt to new diseases and environmental conditions, making them robust and versatile. The integration of deep learning-based disease detection systems in smart farming technologies is poised to revolutionize plant health monitoring, providing farmers with valuable insights into potential threats at an early stage (Xia et al., 2019).

Plant disease detection plays a pivotal role in modern agriculture, addressing the critical need to identify early signs of infections that are frequently imperceptible to the naked eye (Jasim & Al-Tuwaijari, 2020). In the dynamic and complex environment of crop cultivation, diseases can silently infiltrate plant populations, causing substantial damage before visible symptoms manifest. Traditional methods of disease detection often rely on man-

ual observations, which can be time-consuming and subjective (Hernández & López, 2020). Deep learning, particularly convolutional neural networks (CNNs), has emerged as a groundbreaking solution to this challenge. By analyzing high-resolution images of plants, deep learning models can discern subtle variations and anomalies that may indicate the onset of a disease. The ability of these models to learn hierarchical representations allows them to capture intricate patterns, making them adept at identifying early signs of infection that might be overlooked by human observers. As a result, the integration of deep learning in plant disease detection not only expedites the identification process but also enables proactive measures to be taken, minimizing the impact of diseases on crop yields and promoting sustainable agriculture practices. Table xx show a comparison of seven research works that focuses on plant disease detection and identification (Liang et al., 2019; Liu & Wang, 2020; Afzaal et al., 2021; Picon, Alvarez-Gila, et al., 2019; D. Li et al., 2020; Yao et al., 2022; D. Shah et al., 2022).

Beyond mere detection and classification, accurately assessing the **severity of a plant disease** is crucial for implementing appropriate control measures (Esgario et al., 2020; K. Li et al., 2022; C. Wang et al., 2021; J. Gao et al., 2021; Xiang et al., 2021). Cui et al. (Cui et al., 2022) examined those deep learning models, equipped with the ability to comprehend intricate patterns and features in images, have shown promise in estimating disease severity. These models can analyze factors such as lesion size, color, and distribution to quantify the extent of damage inflicted on plants (Wei, Johnson, Langston Jr, Mehl, & Li, 2021). Hernández and López (Hernández & López, 2020) identified that the incorporation of deep learning in severity estimation not only enhances precision but also reduces the subjectivity associated with visual assessments. Furthermore, the integration of other sensing technologies, such as hyperspectral imaging, alongside deep learning, has provided a holistic approach to disease severity estimation (Nguyen

et al., 2021; Pane, Manganiello, Nicastro, & Carotenuto, 2021). The combination of spectral information and deep learning algorithms allows for a more comprehensive analysis of plant health, leading to more informed decision-making in agriculture (Dhiman et al., 2022).

After reviewing the literature, we found that researchers seldom use clustering or deep clustering algorithms for **categorizing plant diseases**. And this motivates us to experiment in this field. Other than the work done by Gokulnath and Usha Devi (Gokulnath & Usha Devi, 2020), none of the presented experiments used deep clustering techniques to tackle plant disease assessment tasks. Gokulnath and Usha Devi (Gokulnath & Usha Devi, 2020) proposed an algorithm that categorizes plant diseases using deep clustering based on DEPICT. In their proposed Boosted-DEPICT model, the embedded learning representation of images is improved by integrating the convolutional autoencoder model with locality-preserving constraints and group sparsity. The model is built using a subset of PlantVillage and PDD image databases. The accuracy of boosted-DEPICT is 97.73% on PlantVillage datasets and 91.25% on PDD datasets, respectively. Finally, several reviews and surveys provided more detailed information about the different applications of images and DL methods in plant disease assessment. Table 2.5 lists some related secondary research efforts.

2.8 Summary

In this chapter, we have deliberated on and compared research endeavors using various criteria, encompassing the employed techniques, the application of ML and DL techniques in agriculture, and the precision of obtained results. Additionally, we have conducted a gap analysis within the existing literature.

Table 2.5: Related secondary research work

<i>#</i>	<i>Ref.</i>	<i>Year</i>	<i>Description</i>
1	(Fang & Ramasamy, 2015)	2015	reviews the direct and indirect disease identification methods
2	(AlKameli & Hammad, 2021)	2021	reviews ML applications in agriculture
3	(L. Li et al., 2021)	2021	reviews DL models used for plant diseases
4	(Liu & Wang, 2021)	2021	reviews DL models used for plant diseases
5	(Saleem, Potgieter, & Arif, 2019)	2019	reviews DL models used to visualize plant diseases
6	(Thakur, Khanna, Sheorey, & Ojha, 2022)	2022	reviews vision-based ML methods for plant disease
7	(Z. Gao, Luo, Zhang, Lv, & Xu, 2020)	2020	reviews the applications of DL in plant stress imaging

This survey is not exhaustive, given the frequent influx of new literature. Nonetheless, the review underscores that, in comparison to other ML and DL techniques, the agricultural applications of deep clustering are still in their infancy. We posit that deep clustering holds the potential to significantly enhance the autonomy of CEA. In the subsequent chapter, we delve into the methodology employed to leverage deep clustering for addressing the challenge of plant disease detection.

Chapter 3

Methods and Procedures

3.1 Overview

Unsupervised learning is a type of machine learning where the algorithm learns patterns and structures in data without any labeled examples or guidance. The main challenge in unsupervised learning is that there is no clear and direct objective function or goal for the algorithm to optimize.

In supervised learning, the algorithm is provided with labeled examples, where each example is associated with a specific class or category. The objective function in supervised learning is typically to minimize the difference between the predicted output and the true label. This allows the algorithm to learn the relationship between input features and their corresponding labels. However, in unsupervised learning, there are no labels available. The algorithm has to discover similarities in the data solely based on the input features. Without explicit labels, it becomes difficult to define a straightforward objective function that can guide the learning process.

In the context of high-level similarities and semantic labels, it means that unsupervised learning struggles to learn complex and abstract concepts or relationships between data points. For example, in image recognition, unsupervised learning may be able to identify basic visual patterns like edges or textures. Still, it may not be able to recognize higher-level concepts like objects or specific categories without any labeled examples or semantic guidance. Hence, the main challenge in unsupervised learning is the absence of a clear objective function that can guide the learning process to discover high-level similarities or semantic labels.

Existing unsupervised representation learning strategies can be categorized into two main concepts: self-supervision and pseudo-supervision. These concepts provide alternative ways to define an objective function or a learning signal in the absence of labeled examples. Self-supervision creates surrogate tasks or generates artificial labels from the data itself, while pseudo-supervision leverages weak or noisy supervision to guide the learning process. These approaches help in learning meaningful representations without relying on explicit labeled examples.

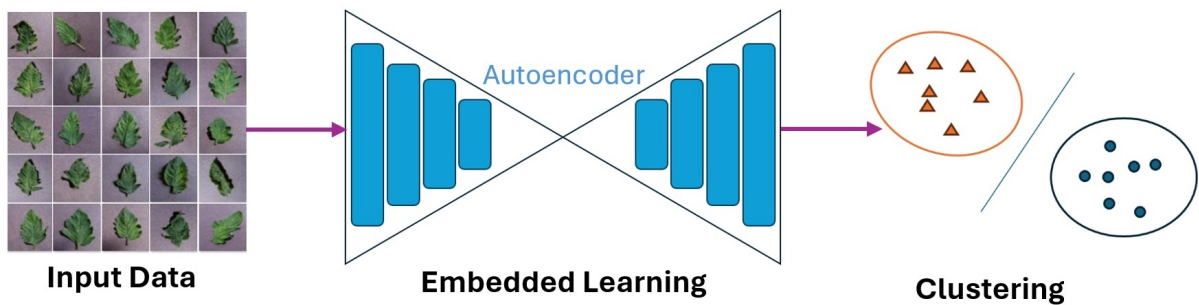


Figure 3.1: Overall framework of the first deep clustering category: embedded learning is conducted prior to clustering

The majority of deep clustering techniques share two fundamental assumptions. The first notion is that deep-embedded representations are more suited to clustering. The second premise is that clustering assignments may be utilized as a supervisory signal while learning embedded representations. Based on this, the existing deep clustering algorithms may be divided into two categories.

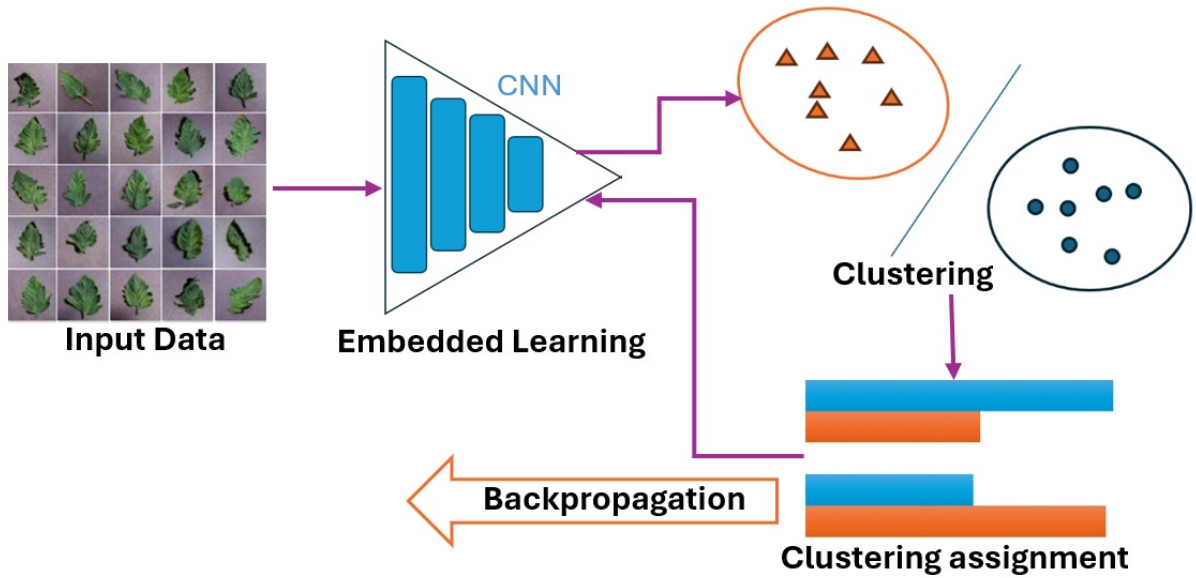


Figure 3.2: Overall framework of the second deep clustering category: simultaneously conducting embedded learning and clustering

The first category distinguishes between embedded learning and clustering. In particular, embedded learning comes before clustering. However, the newly generated labels from the clustering step are not used to train more discriminative features. As shown in figure 3.1, the data is initially projected in a lower-dimensional space using an autoencoder, followed by a clustering of the embedded data. Figure 3.1 shows the two-step framework of this category.

On the other hand, in the other category, embedded learning and clustering are carried out concurrently. Among the most prevalent methodologies used in this category, a conventional clustering algorithm is used to divide the embedded data points into separate groups. Following that, the encoding transformation is trained to drive the embedded space towards a clustering-friendly representation using the previously determined clustering assignment. These two processes can be repeated alternately for a set number of iterations or until a convergence condition is met. Figure 3.2 presents this technique.

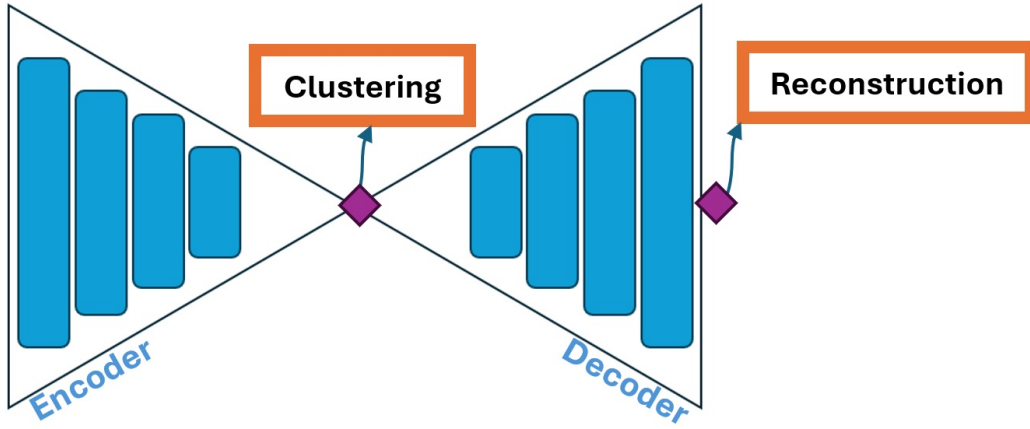


Figure 3.3: The network topology that most autoencoder-based clustering techniques share

This chapter aims to provide a comprehensive overview of the methods and procedures adopted to conduct the research, shedding light on the systematic approach employed in addressing the research questions. The research methodology is a crucial aspect of this study, serving as the roadmap that guides the exploration of deep clustering in the realm of plant disease detection. This chapter is structured as follows. We discuss the trade-off between feature randomness and feature drift, which serves as the impetus for our suggested strategy, in section 3.2. After that, in section 3.3, we introduce our proposed model (DynAE). The other clustering models we are using as a baseline for comparison proposes are introduced in section 3.4. Finally section 3.5 summarizes the chapter.

3.2 The Challenge

Figure 3.3 shows the common network architecture of autoencoder-based clustering techniques. In this multi-objective optimization, trade-off between competing objectives is considered. Objective functions are said to be conflicting if no solution exists that optimizes all of them at the same time. When two objective functions disagree, optimizing one might impair the value of the other. In this scenario, it is feasible to obtain an endless number of Pareto optimum solutions. As a result, more subjective information is necessary to narrow down the options to one. Without this information, all options are equally legitimate.

The clustering reconstruction trade-off refers to the balance between the accuracy of clustering and the ability to reconstruct the original data points. In clustering, the goal is to group similar data points together based on certain criteria. However, in the process of clustering, some information about the original data may be lost. If the clustering algorithm focuses solely on accurately grouping the data points, it may sacrifice the ability to reconstruct the original data accurately. In this case, the clusters may be well-defined and distinct, but the individual data points within each cluster may not represent the true characteristics of the original data accurately. This is known as a high reconstruction error.

On the other hand, if the clustering algorithm prioritizes the ability to reconstruct the original data accurately, it may result in less distinct and more overlapping clusters. In this case, the individual data points within each cluster may better represent the original data, but the boundaries between clusters may become blurred, making it difficult to differentiate between them. This is known as a low clustering accuracy.

Hence, there is a trade-off between achieving high clustering accuracy and low reconstruction error. The choice between prioritizing clustering accuracy or reconstruction depends on the specific application and the goals of the analysis. The clustering reconstruction trade-off is related to feature randomness and feature drift in the context of data analysis and clustering algorithms.

Feature randomness refers to the presence of random or noisy features in the data. These features may not carry meaningful information and can introduce variability that hinders the accuracy of clustering. When there is a high degree of feature randomness, it becomes challenging to accurately group data points based on meaningful patterns. As a result, the clustering algorithm may struggle to reconstruct the original data accurately, leading to a higher reconstruction error. Feature Drift occurs when a neural network is trained to minimize two highly conflicting loss functions simultaneously. Training a neural network with two strongly conflicting objectives has its challenges. The features learned by optimizing the primary objective function can easily be disrupted by updates to the learning weights driven by the secondary objective function. Both feature randomness and feature drift can impact the clustering reconstruction trade-off. If the data has a high degree of randomness or if there are significant drifts in the features, it becomes more challenging to accurately group the data points and reconstruct the original data. This can lead to a trade-off where the clustering accuracy decreases, while the reconstruction error increases. To address this trade-off, it is important to consider techniques that can handle feature randomness and feature drift to remove irrelevant or noisy features, and adaptive clustering algorithms that can adjust to changing patterns in the data. By mitigating the impact of feature randomness and feature drift, it is possible to improve both the clustering accuracy and the reconstruction of the original data.

3.2.1 Feature Randomness

Feature randomness refers to the inherent variability or noise present in the values of features or attributes of data points within a cluster. When performing clustering, each data point is described by a set of features that capture different aspects or characteristics of the data. However, due to various factors such as measurement errors, data collection processes, or natural variability in the data, the values of these features can exhibit randomness or variation within a cluster. Feature randomness can arise from different sources, such as measurement errors, inherent variability, and data preprocessing. Errors or inaccuracies occur during data collection in measuring the values of features. These errors can introduce randomness or noise in the feature values, making them deviate from the true underlying values. In some cases, the data itself may exhibit natural variability or randomness. For example, in biological or environmental datasets, certain features may be influenced by random factors such as genetic variations or environmental fluctuations. This can lead to feature randomness within clusters.

When a neural network is trained using pseudo-supervised labels, it often exhibits feature randomness. This randomness can be characterized at each iteration as the deviation of the gradient of the true supervised objective function from the network parameters, caused by the introduction of pseudo-labels. Mathematically, Feature Randomness is expressed as the cosine of the angle between the gradients of the true supervised objective function and the unsupervised objective function. The characterization of Feature Randomness, denoted as ΔFR , is described by Equation 3.1. This equation quantifies Feature Randomness as the cosine of the angle between the gradient of the loss function with respect to the true labels and the gradient with respect to the pseudo-labels.

$$\Delta FR = \cos \left(\frac{\partial L(x, y_{\text{true}}, w)}{\partial w}, \frac{\partial L(x, y_{\text{pseudo}}, w)}{\partial w} \right) \quad (3.1)$$

where x represents the input data, L is the loss function, y_{true} is the vector of true labels, and y_{pseudo} is the vector of pseudo-labels.

The presence of feature randomness can pose challenges in clustering analysis. It can make it difficult to accurately identify and define the boundaries of clusters since the feature values within a cluster may not be consistent or follow a clear pattern. The randomness can blur the distinction between different clusters, making it harder to separate them accurately. To mitigate the impact of feature randomness, it is important to apply appropriate techniques and methods. This can involve techniques such as outlier detection, data cleaning, or feature selection and using robust clustering algorithms that are less sensitive to noise.

3.2.2 Feature Drift

Modern autoencoder-based clustering algorithms seamlessly integrate representation learning with clustering by optimizing a combined objective that balances the strengths of both approaches. This is typically achieved through the use of a linear combination of two objective functions: one that focuses on learning compact and meaningful representations of the input data (often through minimizing reconstruction error), and another that drives the clustering process (such as minimizing a clustering loss that encourages the formation of distinct clusters). By jointly optimizing these objectives, the model is able to simultaneously learn a low-dimensional embedding that captures the underlying structure of the data and group similar data points

together, leading to more coherent and well-defined clusters. This integrated approach not only enhances the quality of the learned representations but also improves the accuracy and interpretability of the clustering results. The joint optimization process is expressed by Equation 3.2:

$$\mathcal{L} = \mathcal{L}_1 + \gamma \mathcal{L}_2 \quad (3.2)$$

where $\mathcal{L}_1$ is the reconstruction loss function, which evaluates how accurately the autoencoder reconstructs the input data from its lower-dimensional representation. $\mathcal{L}_2$ represents the embedded clustering loss, which encourages the creation of distinct clusters within the latent space. γ is a hyperparameter that balances the two loss functions, determining the trade-off between precise data reconstruction and effective clustering in the embedded space. By tuning γ , the model can be optimized to prioritize either the fidelity of the data representation or the distinctness of the clusters, depending on the specific objectives of the task.

Multi-objective optimization is the process of balancing and optimizing multiple, often conflicting, objectives simultaneously. In such scenarios, the goal is not just to find a single optimal solution, but to navigate trade-offs between competing objectives to identify the best possible compromises. When objective functions conflict, it means that no single solution can optimize all of them to their fullest extent simultaneously. Improving the outcome of one objective may lead to a deterioration in another. This inherent conflict gives rise to a set of solutions known as Pareto-optimal or Pareto-efficient solutions.

Feature Drift occurs when a neural network is trained to minimize a combination of two highly conflicting loss functions. In the specific context of deep clustering, these two loss functions are the pseudo-supervised loss and the self-supervised loss. The pseudo-supervised loss relies on pseudo-labels, which are derived from an initial classification or clustering process. This loss function guides the network to align its representations with these labels, potentially constraining the model’s ability to learn from the underlying data distribution. Conversely, the self-supervised loss emphasizes the network’s ability to learn meaningful representations directly from the data itself, independent of any external labels. Mathematically, Feature Drift can be quantified as the cosine of the angle between the two gradient vectors associated with the self-supervised and pseudo-supervised objective functions. This characterization, denoted as ΔFD , is represented mathematically in Equation 3.3.

$$\Delta FD = \cos \left(\frac{\partial L_P(x, y_{\text{pseudo}}, \hat{w})}{\partial w}, \frac{\partial L_S(x, y_{\text{pretext}}, w)}{\partial w} \right) \quad (3.3)$$

In this equation, L_P represents the pseudo-supervised loss function, with y_{pseudo} being the corresponding pseudo labels and $\hat{w}$ as the estimated parameters. Meanwhile, L_S signifies the self-supervised loss function, where y_{pretext} corresponds to the self-acquired labels for the pretext task, and w are the network parameters.

In summary, feature randomness and feature drift in deep clustering with reconstruction loss can be understood by examining the variability or uncertainty in the reconstructed data compared to the original input data, as captured by the reconstruction loss. Monitoring changes in the reconstruction loss can provide insights into the presence of feature randomness and drift in the deep clustering model. Feature randomness refers to the variation in the values of data points’ characteristics within a given cluster. Cluster-

ing describes each data point with a collection of attributes, the values of which might fluctuate within a cluster owing to noise or intrinsic variability in the data. Cluster borders might be difficult to recognize and define correctly due to feature unpredictability. In contrast, feature drift refers to the level of competition between the gradients of the self-supervised and pseudo-supervised objective functions with respect to the network parameters $\mathbf{w}$. This competition manifests as the interaction between the two gradients, which can either reinforce or undermine each other depending on their respective directions and magnitudes. Understanding this competitive dynamic is crucial for addressing Feature Drift effectively. The coming section describes our double-stage deep clustering model. This section explains the pre-training process and how to apply the dynamic loss function to the plant disease dataset, which is the study’s main contribution.

3.3 Proposed Dynamic Autoencoder Model

As mentioned earlier, in a standard autoencoder-based clustering model, increasing the balancing coefficient substantially leads to the pseudo-supervised objective function overpowering the optimization process. This dominance typically results in an increase in Feature Randomness. Conversely, if the balancing coefficient is reduced excessively, the self-supervised gradient gains prominence, which can undermine the discriminative representations learned through pseudo-supervision. In this scenario, the self-supervised objective function can easily disrupt the learned features, causing undesirable drift. Therefore, the method we are using aims to achieve a careful balance between Feature Randomness and Feature Drift, optimizing both objectives to enhance the overall performance of the clustering model.

Pseudo-labels derived from random latent representations tend to be un-

reliable, as completely random labels do not provide a strong supervisory signal for effective learning. This unreliability leads to excessive Feature Randomness, which can distort the entire learning process. To address this issue, it is beneficial to pretrain the neural network on a relevant pretext task to extract valuable information before incorporating the pseudo-supervised objective function. This pretraining phase helps the network establish a solid foundation, making it more adept at learning discriminative representations when effective pseudo-supervision is introduced. Consequently, any self-supervised loss can serve as a suitable candidate for the pretraining phase, enhancing the overall quality of the learned features. We are using Dynamic Autoencoder (DynAE) (Mrabah, Khan, et al., 2020) in this work, which is a novel deep clustering framework that enhances existing autoencoder designs by introducing dynamic elements. It implements reconstruction regularized by adversarially constrained interpolation and data augmentation due to their impressive performance relative to state-of-the-art self-supervised methods. Fig 3.4 represent the general architecture of the proposed model. It illustrates that DynAE consists of two primary components: the encoder and the decoder. The encoder module maps the input data into a lower-dimensional latent space. In contrast, the decoder module uses the latent representation to reassemble the original input.

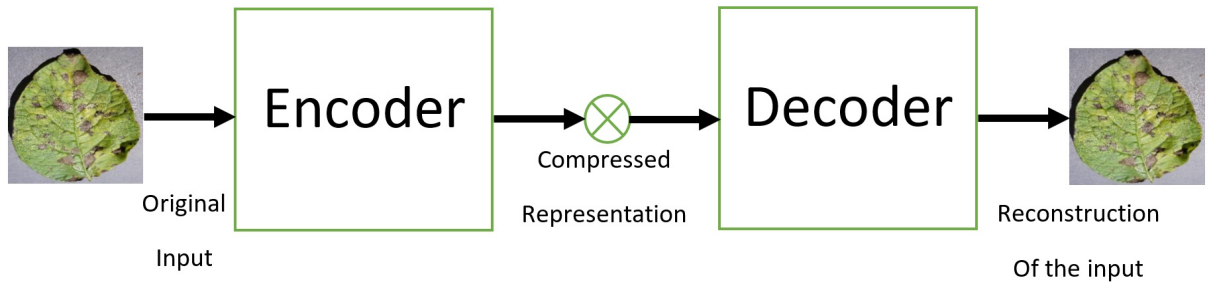


Figure 3.4: dynAE Architecture

We are applying the method of Dynamic Autoencoder to detect plant

diseases. Algorithm 1 illustrates the strategy we are using. Before delving into describing DynAE methodology, we first define some essential notations. We consider the task of clustering a dataset $X = \{x_i \in \mathbb{R}^d\}_{i=1}^N$, which comprises N data points in $\mathbb{R}^d$. This dataset can be organized into K distinct clusters $\{C_k\}_{k=1}^K$. The functions $f_{\theta_e} : X \rightarrow Z$ and $g_{\theta_d} : Z \rightarrow X$ represent the encoder and decoder mappings, respectively, with θ_e and θ_d denoting their learnable parameters. Let $z_i = f_{\theta_e}(x_i) \in \mathbb{R}^p$ denote the latent representation of the data point x_i , while $\hat{x}_i = g_{\theta_d}(z_i) \in \mathbb{R}^d$ signifies the reconstructed representation of x_i . The number of clusters K is treated as prior knowledge, and $\Gamma = \{\mu_j \in \mathbb{R}^p\}_{j=1}^K$ represents the set of latent or embedded centroids. To evaluate the similarity between the latent centroids μ_j and the embedded data points z_i , we utilize the Student's t-distribution as the kernel function. In the algorithm, nC represents the count of conflicted points and nCP represents the count of conflicted points in the previous iteration. Moreover, hyperparameters are generally considered dataset-specific. In our scenario, β_1 and β_2 are determined using mathematical formulas. The algorithm demonstrates their dependence on κ , the confidence threshold. Also, κ is updated in a way that captures the dynamics of the number of conflicted data points.

3.3.1 Pretraining

DynAE, like other autoencoder-based clustering techniques, incorporates a pretraining phase in which both the encoder and decoder are trained to accomplish a specific pretext task. This phase allows for the fine-tuning of the learning weights based on the primary objective function. By leveraging a self-supervised goal during pretraining, the model effectively captures the underlying distributional properties of the data, thereby preventing the extraction of pseudo-labels from random latent representations.

Algorithm 1 Dynamic Autoencoder

Inputs:

pd : Plant disease images dataset,
 K : Number of clusters,
 m : Maximum iterations,
 ϑ : Learning rate,
 t : Convergence threshold,
 κ : Confidence threshold.

Output:

W : Autoencoder weights

Initializations:

Use pretraining strategy to initialize W .
Use kmeans to initialize Γ .
Use $\frac{\kappa}{K}$ to initialize β_1 .
Use $\frac{\beta_1}{2}$ to initialize β_2 .
Use $\text{countConflictedPoints}(pd, \Gamma, \beta_1, \beta_2)$ to initialize nC .
Use $|pd|$ to initialize nCP .

for $i = 0$ **to** m **AND** $\frac{nC}{|pd|} < t$ **do**

if $nC \geq nCP$ **then**

$\Gamma \leftarrow \text{updateCentroids}(pd, K)$

$\beta_1 \leftarrow \beta_1 - \frac{\nabla \kappa}{K}$

$\beta_2 \leftarrow \beta_2 - \frac{\nabla \kappa}{K}$

$\kappa \leftarrow \nabla \kappa$

end if

 Update L using the loss function.

 Update $(W \leftarrow W - \vartheta \frac{\partial L}{\partial W})$.

 Update $(nCP \leftarrow nC)$.

 Update $(nC \leftarrow \text{countConflictedPoints}(pd, \Gamma, \beta_1, \beta_2))$.

end for

In contrast to other autoencoder-based clustering methods, DynAE pre-training phase focuses on optimizing a vanilla reconstruction objective function, enhanced by data augmentation and adversarially constrained interpolation techniques. This innovative approach has demonstrated competitive performance in learning effective unsupervised representations, enabling the model to better capture the underlying structure of the data. This pretraining step is essential for minimizing feature randomness, ensuring that the learned representations are more robust and reliable before transitioning to the main clustering objectives.

Overall, the pretraining phase of the dynamic autoencoder proposed by (Mrabah, Khan, et al., 2020) aims to train the model to learn representations that capture meaningful patterns in the input data, thereby facilitating effective clustering and other unsupervised learning tasks.

3.3.2 Clustering phase

After the pretraining phase, DynAE fine-tune the autoencoder weights by optimizing a dynamic loss function. To initiate this process, we employ K-Means to initialize the embedded clustering centers. For simplicity and to empirically demonstrate the potential of DynAE as a promising deep clustering strategy, we have chosen K-Means for both initializing and updating the centroids. Consistent with state-of-the-art autoencoder-based clustering models, our overall cost function comprises two components: the reconstruction loss and the clustering objective. The reconstruction loss ensures that the model effectively captures the underlying structure of the data, while the clustering objective drives the formation of meaningful clusters within the latent space. Fig 3.5 shows DynAE’s generic clustering design.

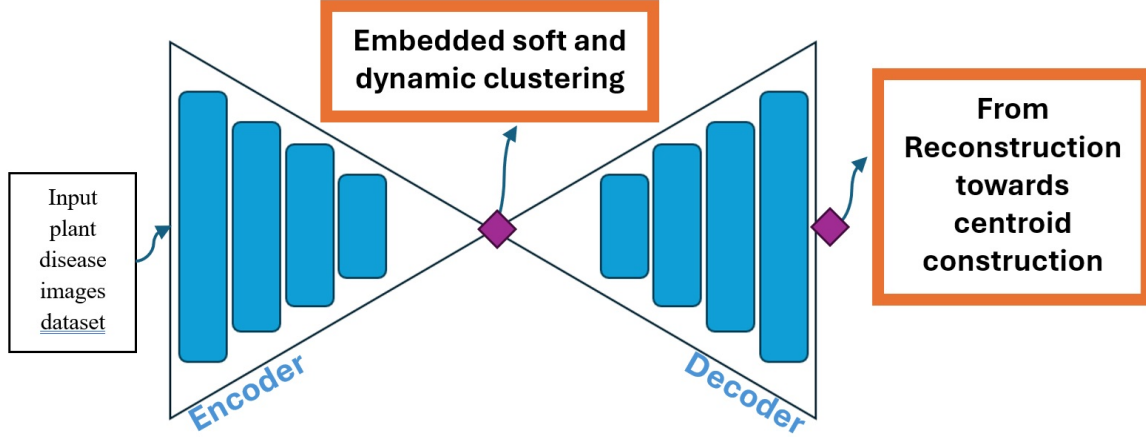


Figure 3.5: Using DynAE clustering architecture to detect plant diseases

DynAE aims to have the autoencoder output the centroid image for each sample, rather than simply generating reconstructed images. To achieve this, the decoder is utilized to produce images of the cluster centroids μ_j from their corresponding embedded representations. These centroid images are then used as supervisory signals fed back into the network. However, abruptly removing the self-supervised reconstruction loss can lead to increased Feature Randomness, while retaining it throughout the entire training process can cause Feature Drift. To address this, a gradual and smooth transition is proposed, where the reconstruction loss is progressively reduced in favor of the centroids construction, thereby balancing the trade-off between Feature Randomness and Feature Drift. Following the pretraining phase, some embedded data points z_i exhibit unreliable clustering assignments q_{ij} . Specifically, the probabilities of their top assignments are nearly indistinguishable, indicating that these samples are likely situated near cluster boundaries. As a result, it becomes challenging to accurately predict their corresponding cluster centers. We refer to these samples as conflicted data points. During each training iteration, we identify an unconflicted data point by applying a condition that regulates the level of acceptable uncertainty in its assignment.

Mathematically, a conflicted data point is defined as belonging to the set S , as outlined in Equation 3.4. Here, β_1 and β_2 are trade-off hyperparameters that range between 0 and 1. Specifically, β_1 represents the minimum confidence threshold below which a data point is considered conflicted, while β_2 is the minimum difference required between the highest and second-highest assignment probabilities (h_{i1} and h_{i2} , respectively). A data point is classified as conflicted if its highest assignment probability is less than β_1 or if the difference between its highest and second-highest assignment probabilities is less than the threshold β_2 .

$$S = \{x_i \in X \mid h_{i1} < \beta_1 \text{ or } (h_{i1} - h_{i2}) < \beta_2\} \quad (3.4)$$

The first component of the dynamic loss function is outlined in Equation 3.5. In this approach, reliable samples are chosen for centroid construction, while conflicted data points continue to be optimized using the reconstruction cost until the model becomes more confident in their corresponding centers.

$$L_1(\theta_e(t), \theta_d(t)) = \sum_{i=1}^N \begin{cases} \|x_i - \hat{x}_i\|_2^2 & \text{if } x_i \in S, \\ \|g(\sigma(x_i)) - \hat{x}_i\|_2^2 & \text{otherwise.} \end{cases} \quad (3.5)$$

Here, x_i is the data point, $\hat{x}_i$ is its reconstructed representation, and S represents the set of conflicted samples. The function $g(\sigma(x_i))$ refers to the centroid construction for reliable samples. Depending on the specific data

sample, the training process follows one of two schemes: reconstruction or centroid construction.

β_1 and β_2 are determined using mathematical formulations, as shown in Equation 3.6. These parameters are dependent on κ , the confidence threshold:

$$\beta_1 = \frac{\kappa}{K} \quad \text{and} \quad \beta_2 = \frac{\beta_1}{2}, \quad \kappa \in [1, K]. \quad (3.6)$$

κ is initialized to ensure that β_1 and β_2 start with relatively high values. As a result, the condition for a data point to be considered unconflicted becomes more stringent. In the early stages of training, this means that a significant portion of the dataset is focused on reconstruction, with only a small subset used for centroid construction. Starting with higher values for κ prioritizes reconstruction over centroid construction, which could slow down the convergence of the model.

κ is adjusted to reflect the changing number of conflicted data points throughout training. The dynamics of the loss function can be described by τ , as shown in Equation 3.7, which represents the proportion of reconstruction in Equation 3.5.

$$\tau = \frac{\text{Number of reconstructed samples}}{N} = \frac{|S|}{N} \quad (3.7)$$

During training, the number of conflicted points is expected to gradually decrease. The dynamic loss function stabilizes once the number of con-

flicted data points no longer declines, at which point τ remains constant.

Equations 3.8 and 3.9 determine the objective function of our suggested embedded clustering. We use the Student's t-distribution for soft clustering assignments, which improves the efficiency of our cost function in handling conflicting data points. The confidence hyperparameters β_1 and β_2 control the learning dynamics, much like they did for the previous loss function in Equation 3.5.

$$L_2(\theta_e(t), \theta_d(t)) = \sum_{x_i \in S} \|f_{\theta_e}(x_i) - \sigma(x_i)\|_2^2, \quad (12) \quad (3.8)$$

$$S = \{x_i \in X \mid h_{i1} \geq \beta_1 \text{ and } (h_{i1} - h_{i2}) \geq \beta_2\}. \quad (13) \quad (3.9)$$

The set of reliable (unconflicted) samples is defined by Equation 3.9, and the number of data points within this set naturally increases with each training iteration, even if β_1 and β_2 remain constant throughout the process.

Finally, Equation 3.10 presents the dynamic objective function in its entirety. Data augmentation is used to regularize the loss function L , same like in the pretraining stage.

$$L_1(\theta_e(t), \theta_d(t)) + L_2(\theta_e(t), \theta_d(t)) \text{ equals } L(\theta_e(t), \theta_d(t)) \quad (3.10)$$

Without depending on any balancing hyperparameters, DynAE aims to achieve the best possible trade-off between Feature Randomness and Feature Drift through its learning dynamics.

3.3.3 Optimization

Mini-batch gradient descent and backpropagation are used to optimize the loss function 3.10 in order to get the autoencoder’s weights. The optimization process is carried out for MaxIter batch iterations or until τ falls below a predetermined percentage of the entire training set. The objective function shifts smoothly and gradually from pure self-supervision to pure pseudo-supervision throughout DynAE training. Initially, the model is pretrained using an objective function that is solely self-supervised. Ultimately, the objective function is nearly entirely pseudo-supervised. Transient objectives are part of this shift from self-supervision to pseudosupervision, where we integrate both approaches. Pseudosupervision rises while self-supervision falls as training goes on.

3.3.4 The main contribution

The main contribution of our thesis is the use of the Gaussian Mixture Model (GMM) instead of the usual K-Means clustering approach in the context of a dynamic autoencoder model. While K-Means is commonly used in conjunction with autoencoders for clustering tasks, its difficulties in dealing with non-spherical clusters and variable densities might restrict the model’s effectiveness, particularly in more complicated datasets. By using GMM, we take use of its capacity to characterize clusters as Gaussian distributions of varying shapes, sizes, and densities, allowing for more flexible and prob-

Table 3.1: Key differences between GMM and K-means

Aspect	K-Means	GMM
Cluster Shape	Spherical	Can be elliptical or any shape
Clustering Type	Hard clustering (each point belongs to one cluster)	Soft clustering (points have probabilities for multiple clusters)
Distance Metric	Euclidean distance	Probabilistic distance (based on Gaussian distributions)
Initialization	Sensitive to initialization (random centroids)	Uses EM algorithm with random initialization of parameters
Convergence	Faster but may converge to local minima	Slower but can find more flexible cluster shapes
Outliers	Sensitive to outliers	Better at handling outliers
Use Case	Works best for simple, spherical clusters	Works best for complex, non-spherical clusters with varying shapes and densities

abilistic data assignment. This enhancement allows the autoencoder model to better capture the data's underlying structure, resulting in more accurate and nuanced cluster assignments. This is especially useful in dynamic and high-dimensional environments where traditional clustering methods such as K-Means may fall short.

GMM and K-means are both popular clustering algorithms, but they differ fundamentally in their approach to partitioning data. K-means is a centroid-based algorithm that assigns each data point to the nearest cluster center, which is iteratively updated to minimize the within-cluster variance. This method assumes that clusters are spherical and of equal size, making it computationally efficient and suitable for well-separated, roughly circular clusters. In contrast, GMM is a probabilistic model that assumes data points are generated from a mixture of several Gaussian distributions, each with its own mean and covariance. Unlike K-means, which assigns each data point to exactly one cluster, GMM assigns a probability that each point belongs to each cluster, allowing for overlapping clusters with varying shapes and sizes. As a result, GMM is more flexible and capable of modeling more complex data distributions but is computationally more intensive. The Expectation-Maximization (EM) algorithm used by GMM involves iterative steps to estimate the parameters of the Gaussian components, which generally makes GMM slower than K-means, particularly in high-dimensional spaces or large datasets. In summary, while K-means is faster and simpler, GMM offers greater flexibility and is better suited for data with more complex underlying distributions. Table 3.1 provides a comprehensive comparison of the key differences between K-means and GMM.

3.4 Baseline Clustering Methods

Understanding the performance criteria established by other techniques is critical when investigating deep clustering for plant disease detection. It is important to assess the effectiveness of our proposed method in contrast to other deep clustering methodologies. Here, we look at other deep clustering algorithms that have been used in image categorization. By providing a baseline, we want to give a framework for evaluating the innovative contributions and gains made possible by deep clustering approaches. This comparative study will not only highlight the potential breakthroughs of our proposed model, but will also help to provide a more nuanced understanding of the context in which these techniques operate.

Deep clustering techniques have the benefit of automatically learning meaningful patterns of data without the need for manual feature engineering. These algorithms can identify complicated patterns that are not immediately evident from raw input data. This makes them especially effective in dealing with high-dimensional and complicated data, like image analysis. In this section, we describe and evaluate five popular deep clustering approaches (Shang, Li, Deng, & Cheong, 2021; Sarfraz et al., 2019; Z. Jiang et al., 2016; X. Guo et al., 2018, 2017) that serve as the foundation of our research. These approaches have garnered significant attention in the field of unsupervised learning due to their effectiveness in partitioning complex data into meaningful clusters. By investigating these techniques, we want to acquire deep insights into their performance and robustness across diverse plant disease datasets for the PDD task.

The **AE + KMeans** (Shang et al., 2021) deep clustering approach combines autoencoders (AE) with KMeans clustering to improve unsupervised learning tasks. Initially, the autoencoder is trained on the input data to de-

velop a compressed representation capable of capturing important information. This pre-trained autoencoder functions as a feature extractor, transforming high-dimensional input into a lower-dimensional latent space. The latent representations created from the autoencoder are then grouped using the KMeans method, which groups data points that are comparable together. This approach partitions complex data into distinct clusters by leveraging autoencoders' non-linear mapping capabilities as well as the simplicity and efficiency of KMeans clustering, providing interpretable and actionable insights for a variety of applications such as image and text analysis, anomaly detection, and customer segmentation. The AE + KMeans approach efficiently performs deep clustering problems by combining the strength of autoencoders with the simplicity of KMeans clustering.

First Integer Neighbor Clustering Hierarchy (FINCH) (Sarfraz et al., 2019) introduces a novel hierarchical clustering approach that relies on integer programming formulations to construct a hierarchy of clusters. It employs an autoencoder architecture to learn a low-dimensional representation of the input while retaining its significant characteristics. FINCH then uses the learnt representation to use a clustering method, like k-means, to group related data points together. FINCH proposes to increase clustering performance by concurrently optimizing the reconstruction loss and clustering objective. At its core, FINCH aims to iteratively merge neighboring clusters based on an integer programming objective function, promoting cluster homogeneity while maintaining a concise and interpretable hierarchy.

Variational Deep Embedding (VaDE) (Z. Jiang et al., 2016) is a powerful generative deep clustering approach that uses Variational Autoencoders (VAEs) and Gaussian Mixture Models (GMMs) to develop a probabilistic representation of the data while also grouping it. It enables the combination

of clustering and data generation. The VaDE methodology entails encoding the initial data distribution in the embedded space as a Gaussian Mixture distribution. The Gaussian Mixture distribution is sampled using the reparameterization approach to facilitate backpropagation. VaDE employs variational inference. As a result, the information loss caused by the mean-field approximation might degrade the quality of the latent space, resulting in poor clustering performance. VaDE introduces a variational lower bound objective that combines the reconstruction loss, clustering objective, and a regularization term. VaDE trains an autoencoder to learn the latent representation of input data. The encoder maps data points to a latent space distribution and the decoder reconstructs data points from the latent space.

Deep Embedding for Clustering (DEC) (X. Guo et al., 2018) extends the notion of autoencoders to learn feature representations suited for clustering by optimizing both an autoencoder and a clustering objective.

The autoencoder and cluster centroids parameters are jointly tuned to minimize the reconstruction error while maximizing the clustering objective, resulting in a compact and discriminative representation of the data suited for clustering. DEC provides cutting-edge performance in a variety of clustering applications using this combined optimization technique, resulting in a robust and scalable unsupervised learning solution. However, DEC lacks a method to prevent Feature Randomness. As a result, using a large number of random labels might result in the generation of random discriminative characteristics. Furthermore, the loss function contribution of each center is normalized to avoid large clusters from influencing the embedded representations. As a result, DEC is more suited to balancing datasets.

Improved Deep Embedded Clustering (IDEC) (X. Guo et al., 2017) improves on the DEC approach by providing a more robust training strategy and including a regularization term to reduce the impact of noisy or irrelevant information. However, IDEC, like DCN, lacks a way to reduce the clustering-reconstruction trade-off. It introduces a soft assignment strategy that allows data points to have multiple cluster assignments with varying degrees of membership. This helps to capture the uncertainty in the clustering process and provides a more flexible representation of the data. IDEC also incorporates a reconstruction loss in addition to the target distribution, which further improves the quality of the learned representation. According to Guo et al. (2017), removing reconstruction during the clustering step hinders the network’s ability to preserve local structures. However, we suggest that because of the intrinsic tension between clustering and reconstruction, retaining the reconstruction loss end-to-end is not as useful. More specifically, embedded clustering ignores non-discriminative patterns that do not help to data classification (for example, between-class similarities), whereas reconstruction is primarily concerned with retaining all information, discriminative or not. The preservation of nondiscriminatory traits from beginning to finish suggests a decline in total discriminative capacity. IDEC, like DEC, uses an autoencoder to develop latent representations of input data suited for clustering.

By maximizing the reconstruction error, the clustering goal, and the regularization term simultaneously, IDEC successfully develops a resilient and compact representation of the data suited for clustering, resulting in enhanced clustering performance and greater generalization to previously unknown data.

3.5 Summary

In brief, deep clustering algorithms utilize deep neural networks to extract hierarchical representations of input data, enabling more effective clustering. By integrating deep learning with clustering techniques, these algorithms uncover intricate patterns and structures within the data, enhancing their utility in unsupervised learning tasks. This synergy between deep learning and clustering empowers the algorithms to capture complex features and relationships, ultimately facilitating the discovery of meaningful clusters in high-dimensional datasets. As a result, deep clustering algorithms play a pivotal role in various domains, offering valuable insights and solutions to challenging unsupervised learning problems. In the upcoming chapter, we endeavor to conduct a comprehensive and thorough evaluation of various deep clustering methods to determine the most effective approach for addressing the specific challenge of plant disease categorization. By systematically comparing the performance, robustness, and scalability of different deep clustering algorithms, we aim to identify the method that yields the most accurate results for this specific critical task.

Chapter 4

Empirical Results and Discussion

4.1 Overview

Through rigorous evaluation and comparison, we aim, in this chapter, to identify the best deep clustering method for addressing the specific challenge of plant disease categorization, thereby contributing to the advancement of unsupervised learning techniques and their practical applications in agricultural machine learning tasks.

The successful identification of an optimal deep clustering method tailored to plant disease categorization stands to revolutionize disease detection and management strategies, ultimately fostering sustainable agricultural practices and ensuring global food security. Our objective is that our efforts not only contribute to the advancement of unsupervised learning techniques but also hold the potential to significantly enhance practical applications in agricultural automation and precision farming.

Our main goal in this chapter is to thoroughly present and evaluate the empirical findings from our investigation. In the next section, we first provide a detailed description of our experimental setup. This covers a detailed explanation of the baseline models that were used, the features of the dataset that was used, the choice of assessment measures, and a summary of the platform that was used for implementations. After that, we describe our experimental results in section 4.3, which offers a comprehensive analysis of the results obtained from our approach. The observed results are then further discussed in section 4.4, where we clarify their significance and consequences in relation to our research goals. In addition, section 4.5 is devoted to examining any challenges to the validity of our thesis, therefore guaranteeing a thorough assessment of our results. In section 4.6, we finally conclude the chapter with a summary that captures the most important discoveries we made during our empirical research.

4.2 Experimental Settings

Here, we outline the main elements of our research technique and we lay the groundwork for our empirical investigation. We first go over the baseline models that were selected for comparison, then we give a summary of the features of the dataset. Next, we go over the evaluation metrics that were chosen with our study goals in mind. In conclusion, we offer a synopsis of the computational infrastructure employed for the implementations. We intend to lay out the methodological foundation for our experimental design and analysis with this concise overview.

4.2.1 Baseline and dataset

The PlantVillage (Hughes, Salathé, et al., 2015) dataset offers a comprehensive depiction of plant diseases for a variety of crop kinds, including cereals, fruits, vegetables, legumes, and decorative plants. The plant species, illness type, severity level, and geographic location are all precisely identified on each image, allowing researchers to investigate the temporal and spatial dynamics of disease prevalence and dispersion. Additionally, the collection includes a range of illness progression stages, from early symptoms to later stages, which makes it easier to design and assess monitoring and diagnostic tools that can identify diseases at different stages of development.

Table 4.1: Plant leaf diseases dataset

Dataset	Number of samples
Potato	2152
Apple	3171
Corn	3852
Grape	4062

The PlantVillage dataset is a valuable resource for advancing the understanding of plant-pathogen interactions and fostering innovation in agricultural technology. It was curated by researchers at Pennsylvania State University in collaboration with contributors worldwide, and it covers a wide range of plant species, diseases, and environmental conditions. We are only utilizing a subset of the PlantVillage dataset. The subset we are using consists of images of plant leaf diseases of four main subsets of the main PlantVillage dataset, which is illustrated in Table 4.1. Moreover, Figure 4.1 shows sample images from the dataset we are using in this study.



Figure 4.1: Plant leaf diseases Dataset (Plant Village)

More description of the five datasets we are using in this study is provided in Table 4.2. These five datasets are subsets of the main PlantVillage dataset.

Table 4.2: Plant leaf diseases dataset

	Crop Type	Class	Instances per class	Total
1	Apple	Healthy	1645	3171
		Apple scab	630	
		Black rot	621	
		Cedar apple rust	275	
2	Corn	Healthy	1162	3852
		Cercospora/Gray leaf spot	513	
		Common rust	1192	
		Northern Leaf Blight	985	
3	Grape	Healthy	423	4062
		Black rot	1180	
		Esca (Black Measles)	1383	
		Leaf blight (Isariopsis Leaf Spot)	1076	
4	Potato	Healthy	152	2152
		Early blight	1000	
		Late blight	1000	

4.2.2 Evaluation Metrics

Clustering assessment measures are critical in determining the quality and performance of clustering algorithms. Among the several metrics available, two popular ones are Accuracy (ACC) (Cai, He, & Han, 2011) and Normalized Mutual Information (NMI) (Strehl & Ghosh, 2002). In this section, we look at the mathematical formulations and intuitive interpretations of these metrics, shining light on their importance in clustering assessment.

ACC is a simple and understandable statistic commonly used in classification problems. However, in a typical clustering problem, because of the lack of ground truth labels, its use to clustering assessment must be approached carefully. In the context of clustering, ACC compares the clustering results to a reference partition or ground truth, if one exists. Let $C = \{C_1, C_2, \dots, C_k\}$ represent the clusters obtained via the clustering method, and $T = \{T_1, T_2, \dots, T_l\}$ represent the genuine clusters. The accuracy is then calculated as the fraction of data points accurately allocated to the same cluster in both the obtained and ground truth clusterings:

$$\text{ACC} = \frac{\sum_{i=1}^k |C_i \cap T_j|}{n}$$

where $|C_i \cap T_j|$ indicates the number of data points shared by cluster C_i and true cluster T_j , for all j , and n is the total number of data points. The ACC runs from 0 to 1, with higher values indicating greater clustering ability.

On the other hand, NMI is a frequently used metric that assesses the similarity of two clusterings by taking into account both clustering agree-

ment and cluster homogeneity. NMI is developed from Mutual Information (MI), which quantifies the amount of information exchanged by two random variables.

Given the obtained clustering C and the true clustering T , the NMI is defined as:

$$\text{NMI}(C, T) = \frac{I(C; T)}{\sqrt{H(C) \cdot H(T)}}$$

where $I(C; T)$ denotes Mutual Information between C and T , and $H(C)$ and $H(T)$ signify the entropies of C and T , respectively. NMI scores range from 0 to 1, with higher values suggesting a stronger resemblance between the two clusters.

ACC and NMI offer useful insights about the performance of clustering methods from several angles. While ACC focuses on the validity of clustering assignments in relation to a ground truth, NMI measures overall agreement and consistency between two clusterings. However, it is vital to recognize that both measurements have limits. ACC can be deceptive when dealing with imbalanced cluster sizes, whereas NMI prefers partitions with a large number of clusters.

To acquire a thorough knowledge of clustering performance, it is advised that different assessment criteria be used in combination. ACC and NMI are crucial methods for evaluating clustering algorithms, providing complementary viewpoints on clustering quality.

Table 4.3: Potato Dataset - DEC, IDEC and DynAE Hyperparameters

	DEC	IDEC	DynAE
X (input data)	2152	2152	2152
maxIter_pretrain	250	250	250
maxIter_cluster	250	500	3000
n_clusters	3	3	3
optimizer (pretrain)	Adam(0.0001)	Dec= Adam(0.0001) IDEC= Adam(0.0001)	Adam(0.0001) avec interpolation
optimizer (train)	-	-	Adam(0.0001)
augmentation pretrain	True	True	True
augmentation cluster	True	True	True
verbose	1	1	1
learning rate (pretrain)	0.0001	Dec (lr=0.0001) IDEC (lr=0.0001)	0.0001
learning rate (cluster)	-	-	0.0001
batch_size	70	70	70
tol (convergence threshold)	-1.	-1.	-1.
loss_weight	-	loss_weight= [1., 0.1]	loss_weight_lambda = 0.5
alpha	1.0	1.0	1.0
Width	0.1	0.1	0.1
Heights	0.1	0.1	0.1
Random rotation	10	10	15
Scale	1.	1.	0.
Gamma	-	-	100

Table 4.4: Apple Dataset - DEC, IDEC and DynAE Hyperparameters

	DEC	IDEC	DynAE
X (input data)	3171	3171	3171
maxIter_pretrain	250	250	250
maxIter_cluster	250	300	500
n_clusters	4	4	4
optimizer (pretrain)	Adam(0.0001)	Dec= Adam(0.0001) IDEC= Adam(0.0001)	Adam(0.0001) avec interpolation
optimizer (train)	-	-	Adam(0.0001)
augmentation pretrain	True	True	True
augmentation cluster	True	True	True
verbose	1	1	1
learning rate (pretrain)	0.0001	Dec (lr=0.0001) IDEC (lr=0.0001)	0.0001
learning rate (cluster)	-	-	0.0001
batch_size	70	70	70
tol (convergence threshold)	-1.	-1.	-1.
loss_weight	-	loss_weight= [1., 0.1]	loss_weight_lambda = 0.5
alpha	1.0	1.0	1.0
Width	0.1	0.1	0.1
Heights	0.1	0.1	0.1
Random rotation	10	10	15
Scale	1.	1.	0.
Gamma	-	-	100

Table 4.5: Corn Dataset - DEC, IDEC and DynAE Hyperparameters

	DEC	IDEC	DynAE
X (input data)	3852	3852	3852
maxIter_pretrain	250	250	300
maxIter_cluster	250	500	1200
n_clusters	4	4	4
optimizer (pretrain)	Adam(0.0001)	Dec= Adam(0.0001) IDEC= Adam(0.0001)	Adam(0.0001) avec interpolation
optimizer (train)	-	-	Adam(0.0001)
augmentation pretrain	True	True	True
augmentation cluster	True	True	True
verbose	1	1	1
learning rate (pretrain)	0.0001	Dec (lr=0.0001) IDEC (lr=0.0001)	0.0001
learning rate (cluster)	-	-	0.0001
batch_size	90	90	90
tol (convergence threshold)	-1.	-1.	-1.
loss_weight	-	loss_weight= [1., 0.1]	loss_weight_lambda = 0.5
alpha	1.0	1.0	1.0
Width	0.1	0.1	0.1
Heights	0.1	0.1	0.1
Random rotation	10	10	15
Scale	1.	1.	0.
Gamma	-	-	100

Table 4.6: Grape Dataset - DEC, IDEC and DynAE Hyperparameters

	DEC	IDEC	DynAE
X (input data)	4062	4062	4062
maxIter_pretrain	200	300	300
maxIter_cluster	250	500	500
n_clusters	4	4	4
optimizer (pretrain)	Adam(0.0001)	Dec= Adam(0.0001) IDEC= Adam(0.0001)	Adam(0.0001) avec interpolation
optimizer (train)	-	-	Adam(0.0001)
augmentation pretrain	True	True	True
augmentation cluster	True	True	True
verbose	1	1	1
learning rate (pretrain)	0.0001	Dec (lr=0.0001) IDEC (lr=0.0001)	0.0001
learning rate (cluster)	-	-	0.0001
batch_size	90	90	90
tol (convergence threshold)	-1.	-1.	-1.
loss_weight	-	loss_weight= [1., 0.1]	loss_weight_lambda = 0.5
alpha	1.0	1.0	1.0
Width	0.1	0.1	0.1
Heights	0.1	0.1	0.1
Random rotation	10	10	15
Scale	1.	1.	0.
Gamma	-	-	100

4.2.3 Implementation and platform

All experiments were carried out on Google Colaboratory platform ¹. All compared models have been implemented in python. The DynAE model's code can be found on Github ².

Moreover, Tables 4.3, 4.4, 4.5 and 4.6 show the hyperparameters settings used in our model for each of the methods DEC, IDEC, and DynAE; for the four datasets for potato, apple, corn and grape, respectively. Also, for DynAE, the values for the hyperparameters kappa (κ), beta 1 (β_1), and beta 2 (β_2) are set to 1, 0.25, and 0.125, respectively.

4.3 Experimental Results

This section presents the empirical basis for evaluating the effectiveness and efficiency of the suggested approach. We provide, in this section, a thorough review of the experimental results obtained by using DynAE and other deep clustering algorithms for the diagnosis of plant diseases. We seek to clarify the benefits, drawbacks, and relative advantages of the suggested methods for precisely and effectively identifying different plant diseases by thorough testing and assessment. It sheds light on the usefulness and possible ramifications of using deep clustering techniques in the field of agricultural disease diagnosis and control.

The performance and applicability of five different deep clustering methods is compared with DynAE (GMM): (1) autoencoder with KMeans, (2) FINCH, (3) VADE, (4) DEC, and (5) IDEC. Table 4.7 provides a comparison of the clustering performances in terms of ACC and NMI.

¹<https://colab.research.google.com>

²<https://github.com/nairouz/DynAE>

Table 4.7: Comparison of the clustering performances in terms of ACC and NMI. Best method in bold

	Potato		Apple		Corn		Grape	
	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
AE+Kmeans	50.41%	20.74%	38.51%	08.98%	59.81%	53.94%	33.21%	04.09%
FINCH	27.60%	03.09%	36.83%	10.86%	51.61%	35.41%	35.99%	05.65%
VADE	50.87%	03.14%	39.57%	10.69%	62.49%	53.99%	34.04%	01.02%
DEC	51.30%	21.81%	41.66%	13.78%	65.58%	54.24%	34.10%	08.01%
IDEC	42.24%	02.78%	40.93%	09.72%	65.97%	60.71%	35.35%	08.93%
DynAE (Kmeans)	72.21%	23.30%	43.01%	10.63%	69.86%	63.37%	44.06%	10.50%
DynAE (GMM)	75.72%	30.59%	45.41%	14.94%	78.99%	63.88%	54.37%	23.65%

A dynamic autoencoder can adaptively improve the quality of its clustering representation and may be more flexible when dealing with different types of data distributions. End-to-end learning and improved clustering accuracy are made possible by DEC and IDEC, which include clustering objectives straight into the neural network training process. VADE, on the other hand, facilitates resilience against noise and uncertainty and allows for probabilistic modeling of latent representations. By using FINCH, we can leverage both local and global structure to achieve better cluster separation by combining feature-wise clustering with hierarchical clustering. On the other hand, autoencoder combined with KMeans is based on a two-phase procedure wherein an autoencoder learns a compressed representation first, and then KMeans is applied for clustering. This method offers simplicity, but it may compromise optimal coherence. To determine their relative merits and demerits in the context of plant disease identification, these approaches can be empirically evaluated and compared in terms of clustering accuracy, scalability, robustness to noise, and computational efficiency. Table 4.7 shows that, when compared to all the other five deep clustering models, DynAE with kmeans continues to produce better results in terms of ACC and NMI. This is evident when comparing our ablated model with the other deep clustering techniques. Furthermore, replacing the K-means clustering algorithm with GMM in the DynAE framework resulted in a significant performance boost. With GMM, DynAE achieved an impressive accuracy of 78.99% on the corn dataset, along with a slight improvement in NMI, reaching 63.88% for the same dataset.

Table 4.8 provides ablation study of DynAE. p_i indicates the i th phase. In the context of deep clustering, an ablation research is methodically deleting or altering model elements in order to determine how each one affects

overall performance. Through this procedure, researchers can identify the critical components that underpin the model’s efficacy and gain insight into which parts are necessary to achieve the best clustering outcomes. Conducting ablation experiments is essential for determining the effect of various design choices, such as network designs, loss functions, or hyperparameters, on the clustering performance in dviseep clustering, where complex neural network architectures are frequently used. Ablation studies offer important insights into the underlying mechanisms governing deep clustering algorithms, directing the improvement and optimization of these techniques for real-world applications by methodically breaking down the model’s architecture and assessing its performance under different configurations.

Table 4.8: Ablation study of DynAE (Kmeans) / DynAE (GMM) : p_i indicates the i th phase

Dataset	Metrics	p_1	$p_1 \ \& \ p_2$	$p_1 \ \& \ p_2$
Potato	ACC	54.97%	73.20%	75.72%
	NMI	20.47%	23.30%	30.59%
Apple	ACC	34.85%	43.01%	45.41%
	NMI	10.40%	10.63%	14.94%
Corn	ACC	68.99%	69.86%	78.99%
	NMI	62.68%	63.37%	63.88%
Grape	ACC	40.13%	44.06%	54.37%
	NMI	09.11%	10.50%	23.65%

We compare the execution time of our model with that of the original DynAE with kmeans, DEC and IDEC in order to more accurately assess its efficiency. We see the following using Table 4.9 which provides a comparison of the execution times of the three approaches in seconds. First, compared to DynAE (Kmeans), DEC and IDEC, respectively, the run-times of DynAE (Kmeans), DEC and IDEC are substantially shorter. Therefore, we can say that overhead is caused by the adversarial pretraining phase. Sec-

Table 4.9: Comparison of the execution times (in seconds) of different deep clustering approaches

	Potato	Apple	Corn	Grape
DEC	205.143	240.090	264.820	283.104
IDEC	211.238	273.750	278.283	242.012
DynAE(Kmeans)	285.562	318.551	328.196	434.821
DynAE(GMM)	557.804	685.141	717.015	896.345

ondly, we note that on all datasets, DynAE (Kmeans) run-time is comparable to both DEC and IDEC. Therefore, we can say that there is no appreciable increase in execution time as a result of our dynamic clustering phase. However, when comparing DynAE with K-means to DynAE with GMM, it is clear that GMM introduces additional computational overhead, slowing down the model. This is expected, as GMM is generally slower than K-means due to its greater computational complexity. While K-means only involves calculating cluster centroids and assigning points to the nearest cluster, GMM models each cluster as a Gaussian distribution with varying shapes and sizes, which requires more complex calculations, including the estimation of covariance matrices and the application of the Expectation-Maximization (EM) algorithm.

Figure 4.2 and Figure 4.3 both illustrate the evolution of ACC and NMI of both DynAE and IDEC during training on Grape dataset. In Figure 4.2, we observe that the average ACC of DynAE is higher than that of IDEC. We also notice from Figure 4.3 that NMI for the DynAE fluctuates and decreases before it starts increasing between the iterations 200 and 250. This is anticipated. The likelihood of making some mistakes is low as long as there are more trustworthy samples available.

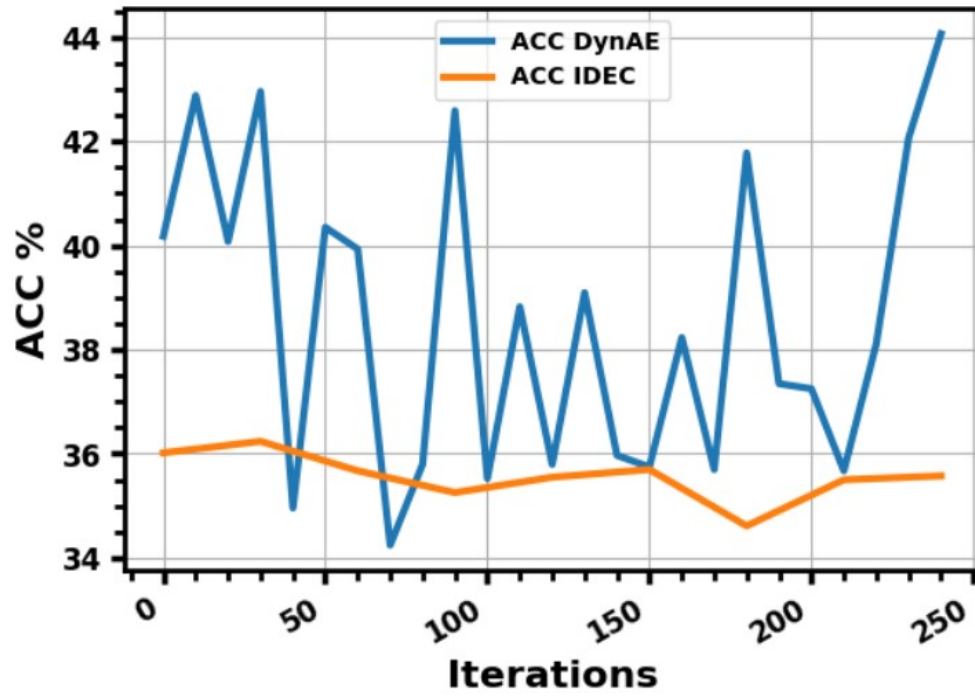


Figure 4.2: ACC between DynAE and IDEC on Grape

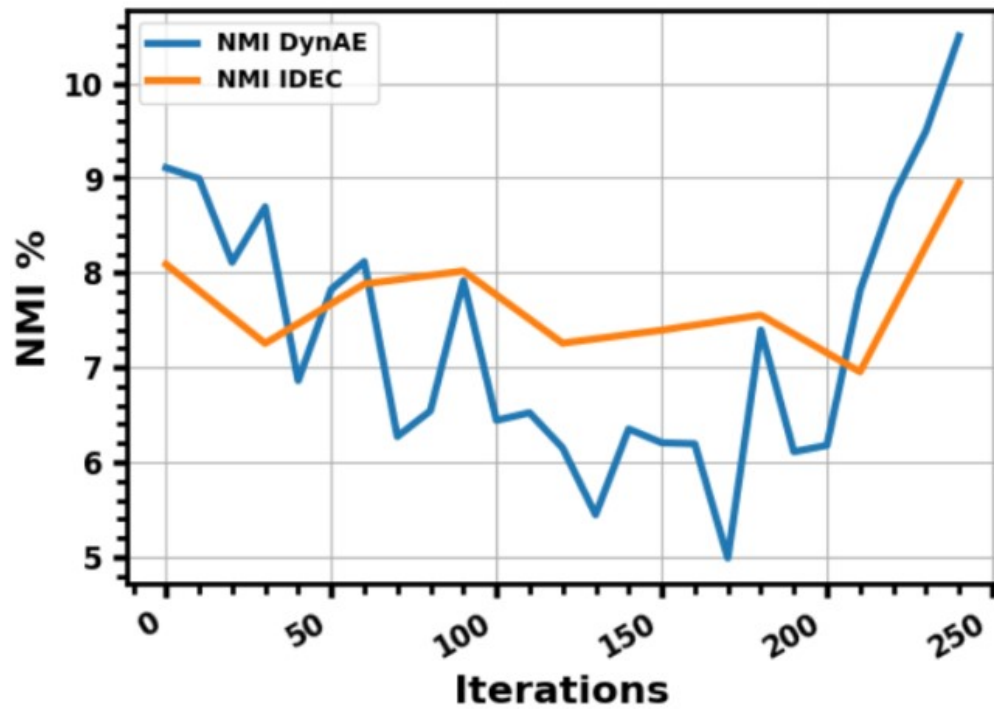
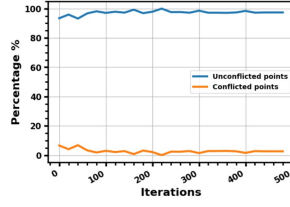
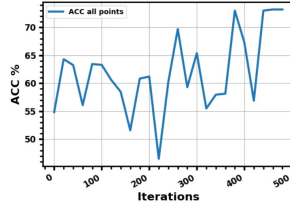


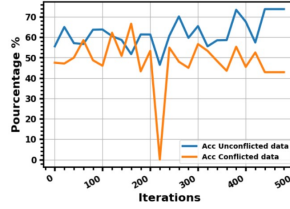
Figure 4.3: NMI between DynAE and IDEC on Grape



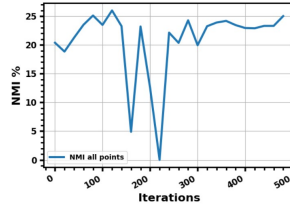
(a) 1a



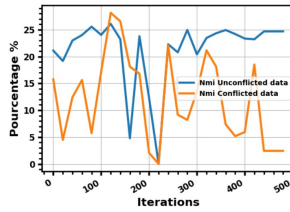
(b) 1b



(c) 1c

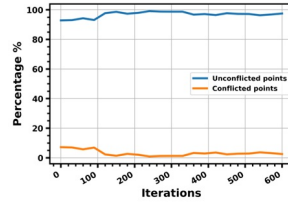


(d) 1d

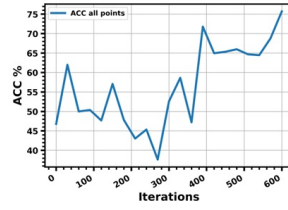


(e) 1e

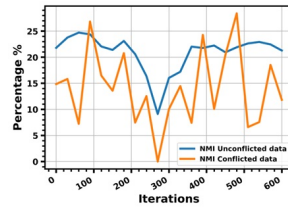
Figure 4.4: Learning dynamics of DynAE (KMeans) on Potato Dataset



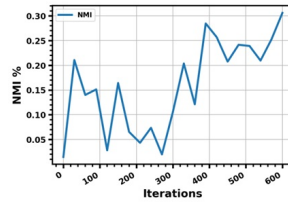
(a) 1a



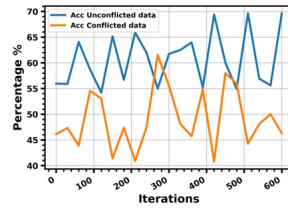
(b) 1b



(c) 1c



(d) 1d

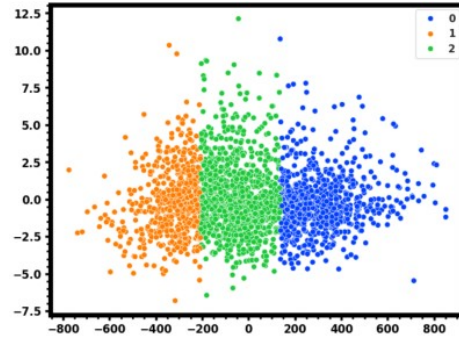


(e) 1e

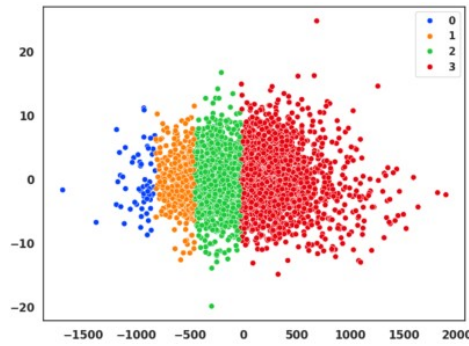
Figure 4.5: Learning dynamics of DynAE (GMM) on Potato

Moreover, points in DynAE can be categorized as "conflicted" or "unconflicted" according to how they behave throughout the clustering process. "Unconflicted points" are data points that the clustering algorithm has confidently assigned to one cluster. These points clearly belong to a certain cluster and are assigned without any doubt or ambiguity. Conversely, "conflicted points" are those whose grouping assignment is characterized by ambiguity or uncertainty. These points might have qualities that are not well-represented by any one cluster, or they might have features that make them borderline cases between numerous groups. Conflicted points therefore present a difficulty to the clustering method since it has to determine which clusters to assign them to, which could produce less trustworthy findings. Figures 4.4 and 4.5 illustrate the learning dynamics of DynAE with K-means and DynAE with GMM on the potato dataset, respectively.

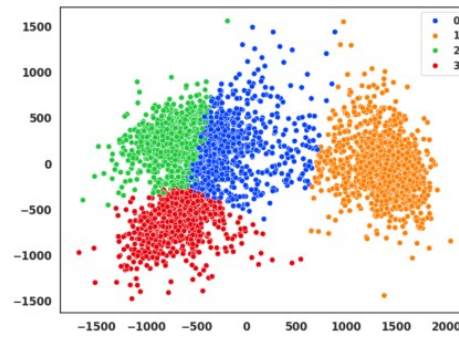
Figure 4.6 illustrates the latent representations learned by the DynAE model through t-distributed Stochastic Neighbor Embedding (t-SNE) visualization. In this plot, each data point represents a latent representation of input samples in a lower-dimensional space, where the distances between points are indicative of their similarities in the original high-dimensional feature space. The t-SNE algorithm effectively preserves the local and global structure of the data, allowing for a visually insightful representation of the clustering patterns discovered by the DynAE model. Different colors or markers may denote distinct clusters or classes identified by the model, providing an intuitive understanding of the clustering performance and the separation between different data clusters. This visualization facilitates the interpretation and assessment of the effectiveness of the DynAE model in capturing underlying data structures and discerning meaningful clusters in complex datasets.



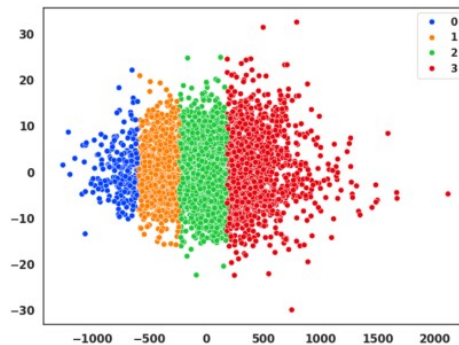
(a) Potato



(b) Apple



(c) Corn



(d) Grape

Figure 4.6: t-SNE visualization of the latent representations of DynAE

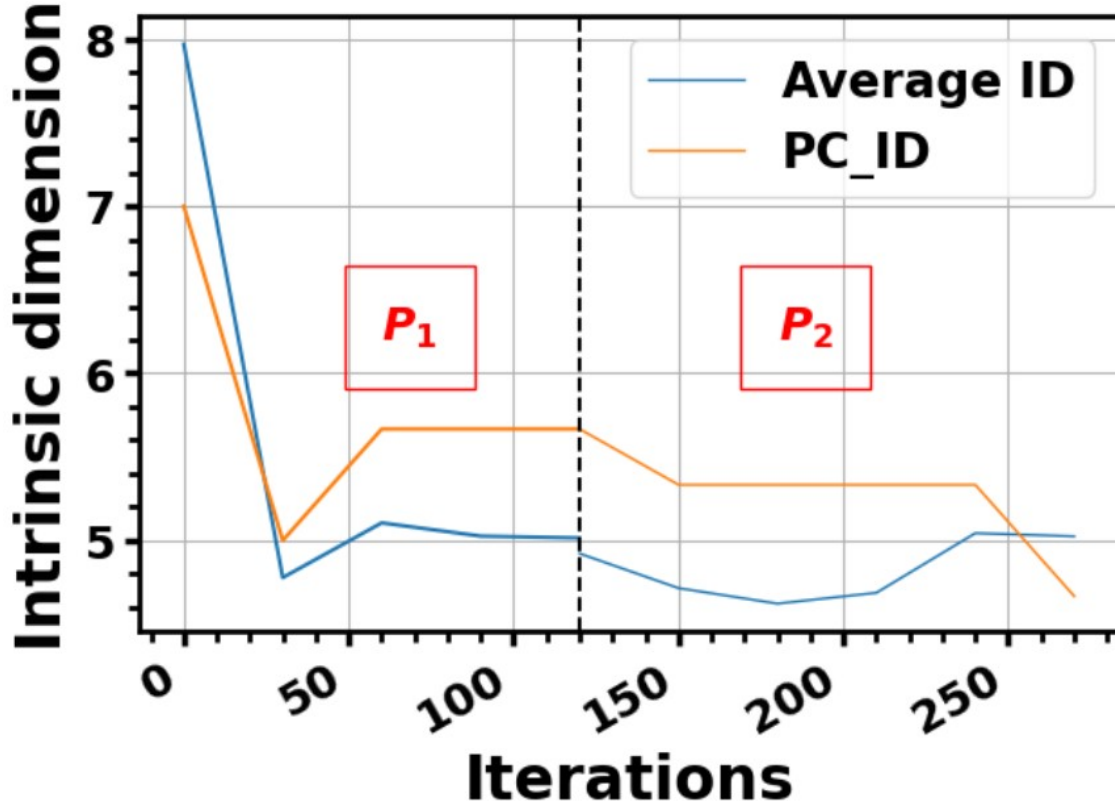


Figure 4.7: Measured intrinsic dimension on Potato (Feature Twist)

Feature twist is a method for estimating a dataset’s intrinsic dimensionality. It entails methodically turning or twisting the feature space and tracking the dataset’s behavior as a result of these changes. The goal is to examine how the structure of the dataset varies when various feature combinations are rotated to emphasize or de-emphasize them. The feature twist approach offers insights into the effective dimensionality of the data by assessing the variation in the dataset’s complexity with different orientations in the feature space. Understanding the dataset’s underlying structure with the help of this information is beneficial, as it can be especially helpful in directing dimensionality reduction methods or determining whether clustering algorithms are appropriate for the given data. Figure 4.7 illustrates the measured intrinsic dimension on potato dataset.

4.4 Discussion

In this section, we address the previously stated research questions after providing the experimental data and analysis. By providing answers to the research questions, we also confirm that the research has been accomplished. In this study, our intention was to develop a novel framework for plant disease categorization using state-of-the-art deep clustering algorithms. We structured the research questions to help achieve our research objectives. The research questions are as follow:

RQ1. How can we utilize ML and DL in improving agricultural activities? — With an emphasis on plant disease identification, this subject is addressed in the literature review via a thorough investigation of ML and DL approaches designed especially for improving agricultural practices. The aim is to investigate revolutionized disease detection and control in agricultural populations by utilizing deep learning architectures. The thesis also explores ML and DL’s potential uses in areas other than disease detection, such as yield prediction, soil analysis, and precision agriculture, highlighting how revolutionary these technologies can be in streamlining agricultural practices and guaranteeing food security for all. By tackling this research subject, the thesis adds to the current conversation about the use of cutting-edge technologies in agriculture and provides creative answers to the intricate problems that face contemporary farming methods. By answering this question, we fulfilled the research objective **RO1**.

RQ2. What are the current trends in plant disease discovery in general and in plant disease clustering? — This subject is fully addressed in the thesis’s literature review, which carefully examines the body of research to clarify current practices, developments, and obstacles in the field of plant disease detection. The review highlights a growing interest in utilizing ma-

chine learning and deep learning approaches for improved disease surveillance and control in agricultural contexts by combining findings from a wide range of studies. In addition, the review carefully examines the particular topic of plant disease clustering, providing insight into the latest advancements and approaches used in this area of study. Highlighting important developments and cutting-edge approaches. By answering this question, we fulfilled the research objective **RO2**.

RQ3. What are the most appropriate clustering methods that can be used to categorize plant disease? — This question is successfully addressed throughout the thesis in accordance with the research objective **RO3**, which is to look into the best clustering techniques for this kind of application. The thesis conducts a thorough empirical inquiry and methodically assesses several clustering strategies, including state-of-the-art deep clustering algorithms and classic methods, for the purpose of classifying plant diseases. Through a thorough examination of the effectiveness, and appropriateness of every technique on plant disease datasets.

RQ4. Can the Dynamic Autoencoder (DynAE) be used to improve plant disease categorization and can it outperform modern clustering models on real benchmark plant disease image datasets? — The analyses carried out in this thesis offer strong support for the central query of whether plant disease classification may be improved by the Dynamic Autoencoder (DynAE). The thesis shows that DynAE is notably effective in enhancing the robustness and accuracy of disease categorization in plant populations through extensive experimental and comparative study. Plant disease data contains complex patterns and latent representations that DynAE is able to capture efficiently by utilizing its hierarchical feature learning capabilities and dynamic update technique. The empirical results also demonstrate DynAE's

improved performance over alternative clustering approaches, suggesting that DynAE could be a useful tool for improving disease detection and control in agriculture. By answering this question, we show how research objective **RO4** is also fulfilled in this research effort. Moreover, using real-world plant disease image datasets, we illustrated that DynAE outperform modern clustering methods. This also partially fulfill the research objective **RO5**.

RQ5. How can we alleviate the two problems of feature drift and feature randomness in plant disease categorization? — By utilizing DynAE, this question is efficiently handled. This method present a viable way to address the problems caused by feature randomization and drift in plant disease categorization applications. By continuously updating their learnt features to match the changing properties of plant disease datasets, dynAEs effectively counteract feature drift by dynamically adapting their internal representations to changes in the input data distribution. Furthermore, DynAEs' hierarchical feature learning skills allow them to identify significant patterns in the data, which effectively lessens the impact of feature randomization and strengthens the resilience of disease classification models. This partially fulfill rsearch objective **RO6**.

RQ6. Can we overcome the clustering-reconstruction trade-off with a dynamic loss function? — The purpose of this thesis inquiry is to evaluate the effectiveness of DynAEs in enhancing both the quality of data reconstruction and clustering performance at the same time. The goal of the research is to thoroughly assess the efficacy of DynAEs in resolving issues such feature drift and feature randomization, which are common in plant disease classification tasks, through research objective **RO6**. We successfully show in our examinations that using a dynamic loss function that can adjust to variations in the input data distribution, DynAEs present a viable

solution to the intrinsic trade-off between reconstruction fidelity and clustering accuracy.

RQ7. How can we improve the DynAE to better solve the plant disease discovery problem? — The thesis attempts to optimize DynAE’s performance in handling complicated, high-dimensional plant disease datasets more efficiently, capturing subtle disease patterns, and adjusting to changing data distributions by methodically examining these enhancements.

While the core structure of the DynAE method was not developed by the candidate, an important contribution to this work was made. Specifically, by replacing the original Kmeans clustering approach with GMM, which enhances the clustering process and addresses certain limitations in the original approach. This modification directly contributes to the thesis’s goal of improving autoencoder-based deep clustering algorithms, as outlined in the research aim, and is applied to the PDD problem. Hence, rather than introducing an entirely new framework, the goal was to refine and tailor existing methods to better suit the specific challenges of the PDD problem. This improvement aligns with the original objectives of the thesis, which focused on addressing the shortcomings of autoencoder-based deep clustering methods.

4.5 Threats to Validity

To ensure the reliability and robustness of the findings given in this thesis, it is critical to identify and address possible issues of validity that may influence the interpretation and generalizability of the results. This section describes these threads and the measures used to reduce their consequences.

Different plant types and diseases may respond differently to deep clustering approaches. Making it difficult to generalize a single approach across plant species and pathogens. Deep clustering methods' efficiency may vary among plant species and plant diseases due to differences in symptom manifestation, biological traits, and environmental variables. The dataset used in this study was carefully chosen to include a variety of plant species and diseases. By including varied plant species such as apple, corn, grape and potato plants, as illustrated in Table 4.2, the suggested methodologies' robustness and generalizability were carefully verified across distinct botanical families and growth patterns.

Also, generally speaking, hyperparameter choices may have an impact on the reproducibility of results when it comes to the performance of deep clustering algorithms. In other words, hyperparameters are understood to be dataset-specific. Furthermore, supervised cross-validation is not feasible for real-world unsupervised applications. Making our technique less susceptible to the selection of sensitive hyperparameters. Here, we compute the hyperparameters β_1 and β_2 using mathematical formulas that depend solely on the confidence threshold, κ , as discussed in the previous chapter. Moreover, deep clustering methods' computing costs may restrict their usefulness, especially for huge datasets. To mitigate that, optimization techniques can be used to increase the suggested approaches' scalability and efficiency. These techniques included mini-batch training, model parallelism, and distributed computing. By using these mitigation tactics, the thesis assures that the deep clustering approaches created for plant disease identification are resilient, increasing their value and application in agricultural and botanical research fields.

4.6 Summary

In this chapter, we outline the experimental framework employed to execute tests across the five chosen datasets. Our study undertook a comprehensive comparison of the novel DynAE (GMM) method against five established deep clustering methodologies: (1) autoencoder with KMeans, (2) FINCH, (3) VADE, (4) DEC, and (5) IDEC. A detailed exposition was provided encompassing the selection criteria and characteristics of the datasets, the metrics employed to gauge performance, and the intricacies of the experimental setup and methodology. Furthermore, we engage in an in-depth analysis of the obtained results, shedding a light on how they effectively address the fundamental research inquiries posited by our thesis. Through experimentation and insightful interpretation, the results discussed in this chapter serves as a pivotal contribution to the advancement of knowledge in the realm of plant disease discovery.

Chapter 5

Conclusion and Recommendations

5.1 Overview

To sum up, this thesis makes a substantial addition to the fields of agricultural disease management and research. We have investigated the use of deep clustering algorithms for the crucial task of diagnosing and treating plant diseases throughout this work. We have shown how cutting-edge machine learning approaches, such as dynamic autoencoders (DynAE), DEC, IDEC, and others, can improve the effectiveness and precision of disease identification in plant populations. Our experimental results highlight how well these methods, especially DynAE, distinguish intricate illness patterns from a variety of datasets, providing encouraging directions for further study and real-world application.

It is essential that we keep improving and fine-tuning these deep clustering techniques as we go along, tackling issues like data heterogeneity, scalability, and practical implementation. In the end, the knowledge gained from

this thesis opens the door for novel approaches to plant disease management, prevention, and surveillance, which have significant effects on agricultural sustainability and global food security.

5.2 Recommendations and Future Work

CEA can provide the perfect environment for the plants but it is not possible without human intervention. To implement fully autonomous greenhouse wireless technology sensors and communication protocols can be used. Robots can be used for seeding, planting, fertilizing, irrigation, weeding, spraying, and harvesting. As a future direction, the research community should aim to study the motivation towards, barriers for, and impact of implementing fully-automated agriculture. According to Jha et al. (Jha et al., 2019), demographic differences of farmers affect their investment in automation. It is worthy, as a starting point, to aim at understanding the actual motivations of farmers toward automation. This will greatly increase the validity of any developed model and improve its usefulness. Interviews and questionnaires can be developed to study the behaviors of farmers and decision-makers and their motivation towards the utilization of computational technology and automation of agriculture. Such studies can help to identify various barriers and dependencies for implementing ML in agriculture. The identification of the driving barriers will help in accelerating the ML implementation. Additionally, by interviewing decision-makers, the environmental, economical, and social impact of utilizing ML in agriculture can be measured. This can provide guidelines on how ML can be deployed to enhance agricultural activities.

5.2.1 General Recommendations related to PDD

We noticed, while surveying the literature, that majority of plant disease research are focusing on one type of disease and one type of plant. We believe, it is worthy to develop models which can combinedly classify multiple numbers of plants, and detect multiple numbers of diseases on different kind of plants. Furthermore, most of the studies focused on leaf and crop disease. A future direction for plant disease could be the recognition of disease on other parts of the plant such as flowers and stems which has not been extensively addressed by researchers. Moreover, research efforts can focus on estimating the severity of the disease.

5.2.2 CEA in the GCC

Future studies should, also, aim at collecting live images from farms in different seasons and incorporate other forms of data such as geographic location, disease history, and weather trends, This can help improve the results of current systems. However, for the settings which are short of data, another direction is to effectively apply transfer learning methods. Cloud computing could, also, be used to improve the learning process efficiency for large-scale learning, especially for complex tasks. Finally, future studies could focus on multi-task and multi-agent learning in smart agriculture.

5.2.3 Deep Clustering for PDD

The ability of deep learning and deep clustering to automatically extract intricate patterns and representations from massive amounts of high-dimensional data makes them excellent approaches for the challenge of plant disease identification. Plant disease datasets, which include spectral data, pictures,

and other multidimensional aspects, are rich and diverse, which complements deep learning models' ability to extract useful information in a hierarchical fashion.

Additionally, deep clustering methods provide a framework for unsupervised learning, making it possible to find hidden patterns and clusters in the data. This is especially helpful in situations when access to labeled data may be limited or costly. Plant disease detection's future rests in developing the combination of deep learning and deep clustering techniques. To do this, new architectures must be investigated, multi-modal data sources must be included, and model interpretability and generalization skills must be improved. Furthermore, the incorporation of deep clustering models and real-time monitoring into agricultural systems has the potential to enable proactive and prompt disease management strategies, protecting global crop yields and guaranteeing food security against changing plant pathogens and environmental stressors.

5.3 The Final Note

Plant diseases are a major danger to global food security, resulting in considerable yield losses and economic impacts. Traditional disease detection methods rely heavily on professional visual inspection, which is time-consuming and prone to human mistake. The application of deep clustering to plant pathology is a viable opportunity for revolutionizing disease finding methods. These approaches use deep neural networks to learn complicated patterns and characteristics from diverse plant datasets, helping to the creation of robust and automated disease detection systems.

In conclusion, the integration of deep learning techniques in plant disease discovery has showcased tremendous potential across various dimen-

sions. From accurate classification and early detection to severity estimation and disease categorization, deep learning has emerged as a transformative force in modern agriculture. The ability of deep learning models to process vast amounts of image data and discern subtle patterns has paved the way for innovative solutions that empower farmers with timely information and insights. As research in this field continues to evolve, the synergy between deep learning and plant pathology holds the promise of enhancing food security and sustainability by mitigating the impact of plant diseases on global crop production.

References

- Abdelbaki, H. H. (2016). Investigation of determinants of groundwater in bahrain. *European Journal of Economics, Finance and Administrative Sciences*(91).
- Afzaal, U., Bhattarai, B., Pandeya, Y. R., & Lee, J. (2021). An instance segmentation model for strawberry diseases based on mask r-cnn. *Sensors*, 21(19), 6565.
- Ahmad, I., Hamid, M., Yousaf, S., Shah, S. T., & Ahmad, M. O. (2020). Optimizing pretrained convolutional neural networks for tomato leaf disease detection. *Complexity*, 2020, 1–6.
- Alameen, A. A., Al-Gaadi, K. A., & Tola, E. (2019). Development and performance evaluation of a control system for variable rate granular fertilizer application. *Computers and Electronics in Agriculture*, 160, 31–39.
- Ale, L., Sheta, A., Li, L., Wang, Y., & Zhang, N. (2019). Deep learning based plant disease detection for smart agriculture. In *2019 ieee globecom workshops (gc wkshps)* (pp. 1–6).
- Alfarisy, A. A., Chen, Q., & Guo, M. (2018). Deep learning based classification for paddy pests & diseases recognition. In *Proceedings of 2018 international conference on mathematics and artificial intelligence* (pp. 21–25).
- Al-Gaadi, K. A., Hassaballa, A. A., Tola, E., Kayad, A. G., Madugundu, R., Alblewi, B., & Assiri, F. (2016). Prediction of potato crop yield using

- precision agriculture techniques. *PloS one*, 11(9), e0162219.
- AlKameli, A., & Hammad, M. (2021). Automatic learning in agriculture: a survey. *International Journal Of Computing and Digital System*.
- Anisi, M. H., Abdul-Salaam, G., & Abdullah, A. H. (2015). A survey of wireless sensor network approaches and their energy consumption for monitoring farm fields in precision agriculture. *Precision Agriculture*, 16(2), 216–238.
- Asaari, M. S. M., Mishra, P., Mertens, S., Dhondt, S., Inzé, D., Wuyts, N., & Scheunders, P. (2018). Close-range hyperspectral image analysis for the early detection of stress responses in individual plants in a high-throughput phenotyping platform. *ISPRS journal of photogrammetry and remote sensing*, 138, 121–138.
- Ball, D., Ross, P., English, A., Patten, T., Upcroft, B., Fitch, R., . . . Corke, P. (2015). Robotics for sustainable broad-acre agriculture. In *Field and service robotics* (pp. 439–453).
- Barbedo, J. G. (2022). Deep learning applied to plant pathology: the problem of data representativeness. *Tropical Plant Pathology*, 47(1), 85–94.
- Barbedo, J. G. A. (2019). Detection of nutrition deficiencies in plants using proximal images and machine learning: A review. *Computers and Electronics in Agriculture*, 162, 482–492.
- Batra, K., & Gandhi, P. (2019). Recent trends of predictive modeling in protected cultivation. In *2019 international conference on machine learning, big data, cloud and parallel computing (comitcon)* (pp. 580–583).
- Bayram, I. S., Al-Qahtani, M., Saffouri, F., & Koç, M. (2017). Estimating the cost of summer cooling in bahrain. In *2017 9th ieee-gcc conference and exhibition (gccce)* (pp. 1–9).
- Benke, K., & Tomkins, B. (2017). Future food-production systems: ver-

- tical farming and controlled-environment agriculture. *Sustainability: Science, Practice and Policy*, 13(1), 13–26.
- Bhatt, P., Sarangi, S., & Pappula, S. (2017). Comparison of cnn models for application in crop health assessment with participatory sensing. In *2017 ieee global humanitarian technology conference (ghtc)* (pp. 1–7).
- Bi, C., Wang, J., Duan, Y., Fu, B., Kang, J.-R., & Shi, Y. (2022). Mobilenet based apple leaf diseases identification. *Mobile Networks and Applications*, 1–9.
- Bressan, N., Bazzaco, L., Bui, N., Casari, P., Vangelista, L., & Zorzi, M. (2010). The deployment of a smart monitoring system using wireless sensor and actuator networks. In *2010 first ieee international conference on smart grid communications* (pp. 49–54).
- Cai, D., He, X., & Han, J. (2011). Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 23(6), 902–913.
- Caron, M., Bojanowski, P., Joulin, A., & Douze, M. (2018). Deep clustering for unsupervised learning of visual features. *Proceedings of the European Conference on Computer Vision (ECCV)*, 132–149.
- Carvalho, A., O’Mahony, N., Krpalkova, L., Campbell, S., Walsh, J., & Doody, P. (2019). Farming on the edge: Architectural goals. In *2019 8th international conference on agro-geoinformatics (agro-geoinformatics)* (pp. 1–6).
- Chang, J., Wang, L., Meng, G., Xiang, S., & Pan, C. (2017). Deep adaptive image clustering. *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 5879–5887.
- Chowdhury, M. E., Rahman, T., Khandakar, A., Ayari, M. A., Khan, A. U., Khan, M. S., ... Ali, S. H. M. (2021). Automatic and reliable leaf disease detection using deep learning techniques. *AgriEngineering*,

3(2), 294–312.

- Coulibaly, S., Kamsu-Foguem, B., Kamissoko, D., & Traore, D. (2019). Deep neural networks with transfer learning in millet crop images. *Computers in Industry*, 108, 115–120.
- Cui, R., Li, J., Wang, Y., Fang, S., Yu, K., & Zhao, Y. (2022). Hyperspectral imaging coupled with dual-channel convolutional neural network for early detection of apple valsa canker. *Computers and Electronics in Agriculture*, 202, 107411.
- Darwish, M. A., Abdulrahim, H., Mohammed, S., & Mohtar, R. (2016). The role of energy to solve water scarcity in qatar. *Desalination and Water Treatment*, 57(40), 18639–18667.
- Dhiman, P., Kukreja, V., Manoharan, P., Kaur, A., Kamruzzaman, M., Dhaou, I. B., & Iwendi, C. (2022). A novel deep learning model for detection of severity level of the disease in citrus fruits. *Electronics*, 11(3), 495.
- dos Santos Ferreira, A., Freitas, D. M., da Silva, G. G., Pistori, H., & Folhes, M. T. (2019). Unsupervised deep learning and semi-automatic data labeling in weed discrimination. *Computers and Electronics in Agriculture*, 165, 104963.
- Esgario, J. G., Krohling, R. A., & Ventura, J. A. (2020). Deep learning for classification and severity estimation of coffee leaf biotic stress. *Computers and Electronics in Agriculture*, 169, 105162.
- Fang, Y., & Ramasamy, R. P. (2015). Current and prospective methods for plant disease detection. *Biosensors*, 5(3), 537–561.
- Fawakherji, M., Youssef, A., Bloisi, D., Pretto, A., & Nardi, D. (2019). Crop and weeds classification for precision agriculture using context-independent pixel-wise segmentation. In *2019 third ieee international conference on robotic computing (irc)* (pp. 146–152).
- Feng, X., Zhan, Y., Wang, Q., Yang, X., Yu, C., Wang, H., . . . He, Y. (2020).

- Hyperspectral imaging combined with machine learning as a tool to obtain high-throughput plant salt-stress phenotyping. *The Plant Journal*, 101(6), 1448–1461.
- Ferentinos, K. P., Katsoulas, N., Tzounis, A., Bartzanas, T., & Kittas, C. (2017). Wireless sensor networks for greenhouse climate and plant condition assessment. *Biosystems engineering*, 153, 70–81.
- Gao, J., Westergaard, J. C., Sundmark, E. H. R., Bagge, M., Liljeroth, E., & Alexandersson, E. (2021). Automatic late blight lesion recognition and severity quantification based on field imagery of diverse potato genotypes by deep learning. *Knowledge-Based Systems*, 214, 106723.
- Gao, Z., Luo, Z., Zhang, W., Lv, Z., & Xu, Y. (2020). Deep learning application in plant stress imaging: a review. *AgriEngineering*, 2(3), 29.
- Gertphol, S., Chulaka, P., & Changmai, T. (2018). Predictive models for lettuce quality from internet of things-based hydroponic farm. In *2018 22nd international computer science and engineering conference (icsec)* (pp. 1–5).
- Ghasedi Dizaji, K., Herandi, A., Deng, C., Cai, W., & Huang, H. (2017). Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization. In *Proceedings of the ieee international conference on computer vision* (pp. 5736–5745).
- Gokulnath, B., & Usha Devi, G. (2020). Boosted-depict: an effective maize disease categorization framework using deep clustering. *Neural Computing and Applications*, 1–10.
- Guo, X., Gao, L., Liu, X., & Yin, J. (2017). Improved deep embedded clustering with local structure preservation. In *Ijcai* (pp. 1753–1759).
- Guo, X., Zhu, E., Liu, X., & Yin, J. (2018). Deep embedded clustering with data augmentation. In *Asian conference on machine learning*

- (pp. 550–565).
- Guo, Y., Zhang, J., Yin, C., Hu, X., Zou, Y., Xue, Z., & Wang, W. (2020). Plant disease identification based on deep learning algorithm in smart farming. *Discrete Dynamics in Nature and Society*, 2020, 1–11.
- Habaragamuwa, H., Ogawa, Y., Suzuki, T., Shiigi, T., Ono, M., & Kondo, N. (2018). Detecting greenhouse strawberries (mature and immature), using deep convolutional neural network. *Engineering in Agriculture, Environment and Food*, 11(3), 127–138.
- Haeusser, P., Plapp, J., Golkov, V., Aljalbout, E., & Cremers, D. (2018). Associative deep clustering: Training a classification network with no labels. *Proceedings of the German Conference on Pattern Recognition (GCPR)*.
- He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. In *Proceedings of the ieee international conference on computer vision* (pp. 2961–2969).
- He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 770–778).
- Helfer, G. A., Barbosa, J. L., Costa, A. B. d., Martini, B. G., & Santos, R. d. (2019). A model for productivity and soil fertility prediction oriented to ubiquitous agriculture. In *Proceedings of the 25th brazilian symposium on multimedia and the web* (pp. 489–492).
- Hernández, S., & López, J. L. (2020). Uncertainty quantification for plant disease detection using bayesian deep learning. *Applied Soft Computing*, 96, 106597.
- Hossain, E., Hossain, M. F., & Rahaman, M. A. (2019). A color and texture based approach for the detection and classification of plant leaf disease using knn classifier. In *2019 international conference on electrical, computer and communication engineering (ecce)* (pp. 1–6).

- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... Adam, H. (2017). Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Hu, W., Miyato, T., Tokui, S., Matsumoto, E., & Sugiyama, M. (2017). Learning discrete representations via information maximizing self-augmented training. *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 70, 1558–1567.
- Hughes, D., Salathé, M., et al. (2015). An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*.
- Hungilo, G. G., Emmanuel, G., & Emanuel, A. W. (2019). Image processing techniques for detecting and classification of plant disease: A review. In *Proceedings of the 2019 international conference on intelligent medicine and image processing* (pp. 48–52).
- Jain, L., Vardhan, H., Nishanth, M., & Shylaja, S. (2017). Cloud-based system for supervised classification of plant diseases using convolutional neural networks. In *2017 IEEE International Conference on Cloud Computing in Emerging Markets (CCEM)* (pp. 63–68).
- Jasim, M. A., & Al-Tuwaijari, J. M. (2020). Plant leaf diseases detection and classification using image processing and deep learning techniques. In *2020 International Conference on Computer Science and Software Engineering (CSASE)* (pp. 259–265).
- Jawad, H. M., Nordin, R., Gharghan, S. K., Jawad, A. M., & Ismail, M. (2017). Energy-efficient wireless sensor networks for precision agriculture: A review. *Sensors*, 17(8), 1781.
- Jha, K., Doshi, A., Patel, P., & Shah, M. (2019). A comprehensive review on automation in agriculture using artificial intelligence. *Artificial Intelligence in Agriculture*.

- Jiang, F., Lu, Y., Chen, Y., Cai, D., & Li, G. (2020). Image recognition of four rice leaf diseases based on deep learning and support vector machine. *Computers and Electronics in Agriculture*, 179, 105824.
- Jiang, Z., Zheng, Y., Tan, H., Tang, B., & Zhou, H. (2016). Variational deep embedding: An unsupervised and generative approach to clustering. *arXiv preprint arXiv:1611.05148*.
- Jones Jr, J. B. (2016). *Hydroponics: a practical guide for the soilless grower*. CRC press.
- Joshi, J., Polepally, S., Kumar, P., Samineni, R., Rahul, S., Sumedh, K., . . . Rajapriya, V. (2017). Machine learning based cloud integrated farming. In *Proceedings of the 2017 international conference on machine learning and soft computing* (pp. 1–6).
- Kaburuan, E. R., Jayadi, R., et al. (2019). A design of iot-based monitoring system for intelligence indoor micro-climate horticulture farming in indonesia. *Procedia Computer Science*, 157, 459–464.
- Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and electronics in agriculture*, 147, 70–90.
- Kim, W.-S., Lee, D.-H., & Kim, Y.-J. (2020). Machine vision-based automatic disease symptom detection of onion downy mildew. *Computers and Electronics in Agriculture*, 168, 105099.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84–90.
- Lakkireddy, K., Kasturi, K., & KRS, S. R. (2018). Role of hydroponics and aeroponics in soilless culture in commercial food production. *Research & Reviews: Journal of Agricultural Science and Technology*, 1(3), 1–8.
- LeCun, Y., Bengio, Y., et al. (1995). Convolutional networks for images,

- speech, and time series. *The handbook of brain theory and neural networks*, 3361(10), 1995.
- Lee, S., Lee, M., Jeon, H., & Smith, A. (2019). Bird detection in agriculture environment using image processing and neural network. In *2019 6th international conference on control, decision and information technologies (codit)* (pp. 1658–1663).
- Lee, S. H., Goëau, H., Bonnet, P., & Joly, A. (2020). New perspectives on plant disease characterization based on deep learning. *Computers and Electronics in Agriculture*, 170, 105220.
- Li, D., Wang, R., Xie, C., Liu, L., Zhang, J., Li, R., . . . Liu, W. (2020). A recognition method for rice plant diseases and pests video detection based on deep convolutional neural network. *Sensors*, 20(3), 578.
- Li, K., Lin, J., Liu, J., & Zhao, Y. (2020). Using deep learning for image-based different degrees of ginkgo leaf disease classification. *Information*, 11(2), 95.
- Li, K., Zhang, L., Li, B., Li, S., & Ma, J. (2022). Attention-optimized deeplab v3+ for automatic estimation of cucumber disease severity. *Plant Methods*, 18(1), 1–16.
- Li, L., Zhang, S., & Wang, B. (2021). Plant disease detection and classification by deep learning—a review. *IEEE Access*, 9, 56683–56698.
- Liang, W.-j., Zhang, H., Zhang, G.-f., & Cao, H.-x. (2019). Rice blast disease recognition using a deep convolutional neural network. *Scientific reports*, 9(1), 2869.
- Liu, J., & Wang, X. (2020). Early recognition of tomato gray leaf spot disease based on mobilenetv2-yolov3 model. *Plant Methods*, 16, 1–16.
- Liu, J., & Wang, X. (2021). Plant diseases and pests detection based on deep learning: a review. *Plant Methods*, 17, 1–18.
- Lucas, J. S., Southgate, P. C., & Tucker, C. S. (2019). *Aquaculture: Farming*

- aquatic animals and plants*. Wiley-Blackwell.
- Maxik, F. S., Soler, R. R., Bartine, D. E., Grove, E. K., Yorio, N., & Crawford, A. (2015, September 15). *Illumination and grow light system and associated methods*. Google Patents. (US Patent 9,137,874)
- Mehra, M., Saxena, S., Sankaranarayanan, S., Tom, R. J., & Veeramanikandan, M. (2018). Iot based hydroponics system using deep neural networks. *Computers and electronics in agriculture*, 155, 473–486.
- Miao, F., Zheng, S., & Tao, B. (2019). Crop weed identification system based on convolutional neural network. In *2019 IEEE 2nd International Conference on Electronic Information and Communication Technology (ICEICT)* (pp. 595–598).
- Mrabah, N., Bouguessa, M., & Ksantini, R. (2020). Adversarial deep embedded clustering: on a better trade-off between feature randomness and feature drift. *IEEE Transactions on Knowledge and Data Engineering*.
- Mrabah, N., Khan, N. M., Ksantini, R., & Lachiri, Z. (2020). Deep clustering with a dynamic autoencoder: From reconstruction towards centroids construction. *Neural Networks*, 130, 206–228.
- Mukherjee, A., Misra, S., & Raghuwanshi, N. S. (2019). A survey of unmanned aerial sensing solutions in precision agriculture. *Journal of Network and Computer Applications*, 102461.
- Murugan, D., Garg, A., & Singh, D. (2017). Development of an adaptive approach for precision agriculture monitoring with drone and satellite data. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 10(12), 5322–5328.
- Nagaraju, M., & Chawla, P. (2020). Systematic review of deep learning techniques in plant disease detection. *International journal of system assurance engineering and management*, 11, 547–560.
- Ngugi, L. C., Abelwahab, M., & Abo-Zahhad, M. (2020). Recent advances

- in image processing techniques for automated leaf pest and disease recognition-a review. *Information Processing in Agriculture*.
- Nguyen, C., Sagan, V., Maimaitiyiming, M., Maimaitijiang, M., Bhadra, S., & Kwasniewski, M. T. (2021). Early detection of plant viral disease using hyperspectral imaging and deep learning. *Sensors*, 21(3), 742.
- Ojha, T., Misra, S., & Raghuwanshi, N. S. (2015). Wireless sensor networks for agriculture: The state-of-the-art in practice and future challenges. *Computers and Electronics in Agriculture*, 118, 66–84.
- Pane, C., Manganiello, G., Nicastro, N., & Carotenuto, F. (2021). Early detection of wild rocket tracheofusariosis using hyperspectral image-based machine learning. *Remote Sensing*, 14(1), 84.
- Patil, V., & Al-Gaadi, K. (2012). Precision fertigation in wheat for sustainable agriculture in saudi arabia. In *Proceedings of the 11th international conference on precision agriculture (icpa)*.
- Picon, A., Alvarez-Gila, A., Seitz, M., Ortiz-Barredo, A., Echazarra, J., & Johannes, A. (2019). Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Computers and Electronics in Agriculture*, 161, 280–290.
- Picon, A., Seitz, M., Alvarez-Gila, A., Mohnke, P., Ortiz-Barredo, A., & Echazarra, J. (2019). Crop conditional convolutional neural networks for massive multi-crop plant disease classification over cell phone acquired images taken on real field conditions. *Computers and Electronics in Agriculture*, 167, 105093.
- Pirani, S. I., & Arafat, H. A. (2016). Interplay of food security, agriculture and tourism within gcc countries. *Global Food Security*, 9, 1–9.
- Qiu, R., Yang, C., Moghimi, A., Zhang, M., Steffenson, B. J., & Hirsch, C. D. (2019). Detection of fusarium head blight in wheat using a deep neural network and color imaging. *Remote Sensing*, 11(22), 2658.
- Rakocy, J., Masser, M. P., Losordo, T., et al. (2016). Recirculating aquacul-

ture tank production systems: aquaponics-integrating fish and plant culture.

- Raza, S., Faheem, M., & Guenes, M. (2019). Industrial wireless sensor and actuator networks in industry 4.0: Exploring requirements, protocols, and challenges—a mac survey. *International Journal of Communication Systems*, 32(15), e4074.
- Roberts, R. K., English, B. C., Larson, J. A., Cochran, R. L., Goodman, W. R., Larkin, S. L., ... Reeves, J. M. (2004). Adoption of site-specific information and variable-rate technologies in cotton precision farming. *Journal of Agricultural and Applied Economics*, 36(1), 143–158.
- R Shamshiri, R., Kalantari, F., Ting, K. C., Thorp, K. R., Hameed, I. A., Weltzien, C., ... Shad, Z. M. (2018). Advances in greenhouse automation and controlled environment agriculture: A transition to plant factories and urban agriculture.
- Saleem, M. H., Potgieter, J., & Arif, K. M. (2019). Plant disease detection and classification by deep learning. *Plants*, 8(11), 468.
- Santos, T. T., de Souza, L. L., dos Santos, A. A., & Avila, S. (2020). Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. *Computers and Electronics in Agriculture*, 170, 105247.
- Saravanan, M., & Perepu, S. K. (2018). Focusing social media based analytics for plant diseases in smart agriculture. In *Proceedings of the 5th multidisciplinary international social networks conference* (pp. 1–7).
- Sarfraz, S., Sharma, V., & Stiefelhagen, R. (2019). Efficient parameter-free clustering using first neighbor relations. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 8934–8943).
- Shah, D., Trivedi, V., Sheth, V., Shah, A., & Chauhan, U. (2022). Rests:

- Residual deep interpretable architecture for plant disease detection. *Information Processing in Agriculture*, 9(2), 212–223.
- Shah, S. A., & Koltun, V. (2018). Deep continuous clustering. *arXiv preprint arXiv:1803.01449*.
- Shang, Q., Li, H., Deng, Y., & Cheong, K. H. (2021). Compound credibility for conflicting evidence combination: An autoencoder-k-means approach. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(9), 5602–5610.
- Sharma, R., Kamble, S. S., Gunasekaran, A., Kumar, V., & Kumar, A. (2020). A systematic literature review on machine learning applications for sustainable agriculture supply chain performance. *Computers & Operations Research*, 104926.
- Shinde, S. S., & Kulkarni, M. (2017). Review paper on prediction of crop disease using iot and machine learning. In *2017 international conference on transforming engineering education (ictee)* (pp. 1–4).
- Shuangpeng, Z., Tao, F., & Hong, H. (2019). Farmland recognition of high resolution multispectral remote sensing imagery using deep learning semantic segmentation method. In *Proceedings of the 2019 the international conference on pattern recognition and artificial intelligence* (pp. 33–40).
- Sidike, P., Sagan, V., Maimaitijiang, M., Maimaitiyiming, M., Shakoor, N., Burken, J., ... Fritsch, F. B. (2019). dpen: deep progressively expanded network for mapping heterogeneous agricultural landscape using worldview-3 satellite imagery. *Remote sensing of environment*, 221, 756–772.
- Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Strehl, A., & Ghosh, J. (2002). Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of Machine*

- Learning Research (JMLR)*, 3(Dec), 583–617.
- Suh, H. K., Ijsselmuiden, J., Hofstee, J. W., & van Henten, E. J. (2018). Transfer learning for the classification of sugar beet and volunteer potato under field conditions. *Biosystems Engineering*, 174, 50–65.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 1–9).
- Thakur, P. S., Khanna, P., Sheorey, T., & Ojha, A. (2022). Trends in vision-based machine learning techniques for plant disease identification: A systematic review. *Expert Systems with Applications*, 118117.
- Tola, E., Al-Gaadi, K. A., & Madugundu, R. (2019). Employment of GIS techniques to assess the long-term impact of tillage on the soil organic carbon of agricultural fields under hyper-arid conditions. *PloS one*, 14(2), e0212521.
- Tripathi, M. K., & Maktedar, D. D. (2016). Recent machine learning based approaches for disease detection and classification of agricultural products. In *2016 international conference on computing communication control and automation (iccubea)* (pp. 1–6).
- Tzoreff, E., Kogan, O., & Choukroun, Y. (2018). Deep discriminative latent space for clustering. *arXiv preprint arXiv:1805.10795*.
- Vadivel, R., Parthasarathi, R., Navaneethraj, A., Sridhar, P., KA, M. N., & Karan, S. (2019). Hypaponics-monitoring and controlling using internet of things and machine learning. In *2019 1st international conference on innovations in information and communication technology (iciict)* (pp. 1–6).
- Wan, H., Lu, Z., Qi, W., & Chen, Y. (2020). Plant disease classification using deep learning methods. In *Proceedings of the 4th international conference on machine learning and soft computing* (pp. 5–9).

- Wang, A. X., Tran, C., Desai, N., Lobell, D., & Ermon, S. (2018). Deep transfer learning for crop yield prediction with remote sensing data. In *Proceedings of the 1st acm sigcas conference on computing and sustainable societies* (pp. 1–5).
- Wang, C., Du, P., Wu, H., Li, J., Zhao, C., & Zhu, H. (2021). A cucumber leaf disease severity classification method based on the fusion of deeplabv3+ and u-net. *Computers and Electronics in Agriculture*, 189, 106373.
- Wei, X., Johnson, M. A., Langston Jr, D. B., Mehl, H. L., & Li, S. (2021). Identifying optimal wavelengths as disease signatures using hyperspectral sensor and machine learning. *Remote Sensing*, 13(14), 2833.
- Weiss, M., Jacob, F., & Duveiller, G. (2020). Remote sensing for agricultural applications: A meta-review. *Remote Sensing of Environment*, 236, 111402.
- Xia, J., Cao, H., Yang, Y., Zhang, W., Wan, Q., Xu, L., ... Huang, B. (2019). Detection of waterlogging stress based on hyperspectral images of oilseed rape leaves (*brassica napus* l.).
- Xiang, S., Liang, Q., Sun, W., Zhang, D., & Wang, Y. (2021). L-csms: novel lightweight network for plant disease severity recognition. *Journal of Plant Diseases and Protection*, 128, 557–569.
- Xie, J., Girshick, R., & Farhadi, A. (2016). Unsupervised deep embedding for clustering analysis. *International Conference on Machine Learning (ICML)*, 478–487.
- Xin, Y., & Adler, P. R. (2019). Mapping miscanthus using multi-temporal convolutional neural network and google earth engine. In *Proceedings of the 3rd acm sigspatial international workshop on ai for geographic knowledge discovery* (pp. 81–84).
- Xu, J., Wei, H., Ye, M., & Wang, W. (2019). Research on recognition method of *zanthoxylum armatum* rust based on deep learning. In *Pro-*

- ceedings of the 2019 3rd international conference on computational biology and bioinformatics* (pp. 84–88).
- Yalcin, H. (2017). Plant phenology recognition using deep learning: Deep-pheno. In *2017 6th international conference on agro-geoinformatics* (pp. 1–5).
- Yang, B., Fu, X., Sidiropoulos, N. D., & Hong, M. (2017). Towards k-means-friendly spaces: Simultaneous deep learning and clustering. In *international conference on machine learning* (pp. 3861–3870).
- Yang, J., Parikh, D., & Batra, D. (2016). Joint unsupervised learning of deep representations and image clusters. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 5147–5156).
- Yang, Y., Cao, H., Han, C., Ge, D., Zhang, W., et al. (2018). Visible-near infrared spectrum-based classification of apple chilling injury on cloud computing platform. *Computers and Electronics in Agriculture*, 145, 27–34.
- Yao, J., Wang, Y., Xiang, Y., Yang, J., Zhu, Y., Li, X., ... Gong, G. (2022). Two-stage detection algorithm for kiwifruit leaf diseases based on deep learning. *Plants*, 11(6), 768.
- Zhang, N., Wang, M., & Wang, N. (2002). Precision agriculture—a world-wide overview. *Computers and electronics in agriculture*, 36(2-3), 113–132.
- Zhou, Z., Li, S., & Shao, Y. (2018). Crops classification from sentinel-2a multi-spectral remote sensing images based on convolutional neural networks. In *Igarss 2018-2018 ieee international geoscience and remote sensing symposium* (pp. 5300–5303).

ملخص الدراسة

يعد اكتشاف أمراض النبات أمراً ضرورياً للحفاظ على الأمن الغذائي والإنتاج الزراعي. غالباً ما تكون تعقيدات وتنوع مجموعات بيانات أمراض النبات أكثر مما تستطيع الطرق التقليدية التعامل معه، لذا هناك حاجة إلى أفكار جديدة لتحسين عمليات اكتشاف الأمراض. من أجل التحكم في أمراض النبات ومراقبتها بشكل أفضل، تبحث هذه الأطروحة في استخدام أساليب التجميع العميق لاكتشاف الأمراض. وهي تفعل ذلك من خلال الاستفادة من قدرات تقنيات التعلم الآلي المتطورة.

من خلال استكشاف شامل لخوارزميات التجميع العميق، بما في ذلك المشفرات التلقائية الديناميكية (DynAEs)، والمشفر التلقائي باستخدام KMeans، وتسلسل التجميع الصحيح الأول (FINCH)، والتضمين العميق المتغير (VADE)، والتجميع المضمن العميق (DEC)، والتجميع المضمن العميق المحسن (IDEC)، تدرس الأطروحة فعاليتها في تصنيف أمراض النبات بدقة. توضح فحوصاتنا أنه عند مقارنتها بجميع نماذج التجميع العميق الخمسة الأخرى، تستمر DynAE في تحقيق نتائج أفضل من حيث ACC و NMI. تسلط الدراسة المقارنة الضوء على مزايا وعيوب كل من الطرق المذكورة، وتقدم معلومات ثاقبة حول مدى نجاحها في المواقف الفعلية.

تكشف النتائج الرئيسية لهذه الأطروحة أن DynAE تفوقت بشكل كبير على طرق التجميع العميق الحديثة الأخرى، مثل Autoencoder مع K-Means و FINCH و VADE و DEC و IDEC، عبر أربع مجموعات بيانات مميزة لأمراض النبات - البطاطس والتفاح والذرة والعنب. تم تقييم الأداء باستخدام الدقة (ACC) والمعلومات المتبادلة الطبيعية (NMI) كمقاييس. ومن اللافت للنظر أن DynAE حقق أعلى دقة بنسبة 72.21٪ في مجموعة بيانات البطاطس، مما يشير إلى تفوقه في تجميع نقاط البيانات بشكل صحيح مقارنة بالطرق الأخرى. بالإضافة إلى ذلك، سجل DynAE أعلى NMI بنسبة 63.37٪ في مجموعة بيانات الذرة، مما يدل على فعاليته في الحفاظ على المعلومات المتبادلة بين المجموعات المتوقعة والحقيقية. تؤكد هذه النتائج على قوة DynAE وإمكاناتها كنهج رائد لمهام التجميع في الكشف عن أمراض النبات. بعد تحليل الأداء العام لـ DynAE، كان التحسين المهم في هذا العمل هو استبدال خوارزمية التجميع K-means بنموذج الخليط الغاوسي (GMM) في إطار عمل DynAE. اعتمدت طريقة DynAE الأصلية على K-means للتجميع، لكن هذه الأطروحة توضح أن GMM يحسن بشكل كبير دقة التجميع وقوته. حقق DynAE مع GMM دقة أعلى بنسبة 78.99٪ على مجموعة بيانات الذرة، جنباً إلى جنب مع NMI محسّن قليلاً بنسبة 63.88٪ على نفس مجموعة البيانات. يؤدي هذا التعديل إلى أداء تجميع متفوق، مع تفوق DynAE باستخدام GMM ليس فقط على نموذج DynAE الأصلي مع K-means، ولكن أيضاً على طرق التجميع العميقة الأخرى.

جامعة البحرين
كلية تقنية المعلومات



تقنيات التجميع العميق لاكتشاف أمراض النبات

أطروحة مقدمة كجزء من متطلبات الحصول على
درجة الدكتوراة في علوم الحوسبة والمعلومات

إعداد

عالية السيد حسين الكاملي

١٩٩٩٠٢٢١

إشراف

د. رياض محمد قسنطيني

(أستاذ مشارك)

د. هشام محمد العمال

(أستاذ مساعد)

جامعة البحرين

مملكة البحرين

نوفمبر ٢٠٢٤