



Deep neural network-driven precision agriculture multi-path multi-hop noisy plant image data transmission and plant disease detection

Derek K. P. Asiedu¹ · Kwabena E. Bennin² · Dennis A. N. Gookyi³ · Mustapha Benjillali^{1,4} · Samir Saoudi¹

Received: 28 June 2024 / Accepted: 19 March 2025
© Institut Mines-Télécom and Springer Nature Switzerland AG 2025

Abstract

Precision agriculture (PA) and plant disease detection (PDD) are essential for farm crops' life quality and crop yield. Unfortunately, current PDD algorithms are trained and deployed with perfect plant images. This is impractical since PA sensor networks (PANs) transfer imperfect data due to wireless communication imperfections, such as channel estimation and noise, as well as hardware imperfections and noise. To capture the influence of channel imperfections and combat its effect, this work considers on- and/or offsite PDD implementation using plant image data transferred over multi-path imperfect PAN. Here, both traditional decode-and-forward (DF) data routing and channel effect considering machine learning data autoencoder multi-path routing are used for image data transmission. The multi-path DF data routing considers equal gain combining (EGC) and maximum ratio combining (MRC) techniques at the destination gateway for data decoding. In addition, a PDD deep learning algorithm is developed to predict whether or not a farm plant is diseased, using the noisy image data captured by the multi-path data routing PAN. From the PAN-PDD integrated system simulation, the proposed ML multi-path PAN-PDD algorithms (i.e., EGC and MRC) are compared to the ML single-path PAN-PDD algorithm and the traditional single-path PAN-PDD system. The simulation results showed that the multi-path approach performed fairly well over the other DF PAN-PDD systems. Incorporating the channel effects in designing an intelligent wireless data transfer solution/technique improves the communication system performance in PDD implementation.

Keywords Deep learning (DL) · Multi-path wireless sensor network (WSN) · Plant disease detection (PDD) · Convolutional neural network (CNN) · Autoencoder

1 Introduction

With the advent of the Internet of Things (IoT) and machine learning (ML), intelligent applications such as smart homes,

smart industry, smart agriculture, and many other applications have been developed and integrated into society to improve the quality of life of individuals within society [1–3]. The application of IoT networks and ML in “smart” agriculture is termed precision agriculture (PA), and the development of its technology to improve farming falls under the recently coined and evolving term Farming 4.0 [1, 2, 4]. Farming 4.0 covers intelligent farm monitoring (e.g., soil nutrients, soil moisture, plant and livestock monitoring, and pests monitoring) and mechanization (e.g., application of fertilizers, feeding of animals, weeding, and spraying), and the intelligent inference and protocol execution (e.g., plant and livestock disease detection, production planning, and farm machinery control). PA seeks to reduce the impact of farming on the climate, make farming more efficient, and satisfy

Kwabena E. Bennin, Dennis A. N. Gookyi, Mustapha Benjillali, and Samir Saoudi contributed equally to this work.

Derek K. P. Asiedu worked on all aspects of the manuscript and research (system model development, coding, simulations). Kwabena E. Bennin and Dennis A. N. Gookyi assisted with the writing and editing of the manuscript and the PDD machine learning algorithm development. Mustapha Benjillali and Samir Saoudi helped with the manuscript edits and the development of the autoencoder design for the communication system.

Extended author information available on the last page of the article

specific United Nations (UN) sustainable development goals (SDGs) set for or associated with modernized farming. The aim is to reduce farming costs, operations, and human labor while increasing the availability of quality food and making farming more attractive to individuals and investors. However, the development of the PA sensor networks (PANs) and the intelligent farm applications (e.g., software and mobile apps) need to be improved for efficiency and to satisfy the UN SDGs [2].

Massive machine-type communications (mMTC) in PAN for farm data acquisition and task execution research and deployment are growing in academic and industrial research [1, 2, 5]. From the protocols and algorithms point of view, ML deep neural networks (DNNs) are increasingly used in PA for farm monitoring and process automation protocols and algorithms [3]. mMTC PANs and DNNs algorithms have been increasingly used in PA research and development plant disease detection (PDD) applications to improve crop production and quality [6, 7]. However, unfortunately, the PAN and DNN PDD have been investigated separately in their research, design, and development stages [4, 5, 7, 8]. This approach is not practical and inefficient. Also, this approach does not factor in the significant influence both systems (PAN and DNN PDD) have on each other. For example, knowing that the PAN acquires and transfers noisy data during data routing due to the farm environment (i.e., the wireless communication channel structure, physical obstructions, and other interfering signals) and the sensor hardware and its operational imperfections. This PAN data transmission imperfection effect is critical and must be factored into the images acquired and used in training DNN PDD applications currently being studied and developed. This is because current DNN PDD algorithms are trained using clean and perfect images, which are impractical compared to the actual PAN noisy image data. In addition, mMTC devices and the current Long Range (LoRA) wireless communication network use a limited number of bits and processing power for transmitting and routing image noisy data over the PAN. Hence, PAN-transferred image sizes must be reduced to allow transmission through farm mMTC devices or use concurrent packet transmissions through the PAN. This strains the PAN, reducing its efficiency and network lifespan. This strain on the PAN reduces the quality of images processed, transferred, and available for PDD. Due to the impractical implementations of current PAN-PDD systems and research, there is a need to improve and integrate the PAN-PDD systems in research and development [9, 10]. This issue has been partially considered in our previous work in [11], which focused on multi-hop single-path PAN-PDD integrated system.

Motivation To further highlight the influence of the PAN on PDD, this work proposes the deployment of an onsite (farm) multi-hop and multi-path DNN PAN using an autoencoder and both the equal gain combining and maximum ratio combining schemes at the destination and also factors in noisy data (image data) transmission due to channel effects. The proposed DNN PAN is combined with a DNN PDD algorithm to mimic a practical integrated PAN-PDD implementation and interaction. The proposed integrated PAN-PDD system can be used for further PAN-PDD investigations in PA. The contributions of this work are as follows.

- i Compared to our previous work in [11], which considered a multi-hop single-path PAN-PDD system model, this work focuses on a multi-hop multi-path PAN-PDD system. Additionally, with this work, both the equal gain combining and maximum ratio combining approaches were considered to combine the multi-path signals at the central system.
- ii Incorporating the negative influence of channel noise on transferred images (distorted images), an autoencoder transceiver design is proposed for the multi-hop multi-path PAN sensor node operation. The autoencoder aids in data size reduction for transmission through the limited resourced routing sensor nodes within the PAN to the on/offsite DNN PDD.¹
- iii This work also presents DNN PDD training, testing, validation, and predictions based on noisy data (crop images) generated from the multi-hop multi-path PAN. Our approach differs from most current research because the existing works used clean captured images, which did not consider the noise introduced during image transmission within a communication network, making the images distorted and containing noise. Hence, their DNN PDD training and implementation uses perfect images for DNN PDD training, testing, and validation. However, our training in this work considered the noisy effects and trained the DNN PDD based on noisy/distorted images.

Finally, through comparison of the proposed multi-hop multi-path DNN PAN-PDD technique (MP-MDF) to the transitional decode-and-forward (DF) multi-hop single-path (SP-TDF) and multi-path (MP-TDF) PAN-PDD implementation showed the superiority of the proposed MP-MDF over the SP-TDF and MP-TDF benchmarks. Also, the multi-path schemes of the MP-MDF (i.e., EGC MP-MDF and MRC MP-MDF) and MP-TDF (i.e., EGC MP-TDF and MRC MP-TDF)

¹ Here, onsite or offsite DNN PDD refers to a centralized system built on or off the farm, respectively.

outperformed their single-path counterparts (i.e., SP-MDF and SP-TDF).

Note, since, this work builds on the work presented in [11], the single-path multi-hop approach is used as a benchmark and for comparison with the multi-path multi-hop scenario presented in this paper. Additionally, to improve readability, Table 1 containing abbreviations is included in the manuscript to be used as a reference point for abbreviations.

2 Precision farming system model

The work presented in this paper investigates an on/offsite DNN PAN-PDD algorithm development based on acquired

Table 1 Summary of abbreviations used

Abbreviation	Definition
2D	Two dimensional
3D	Three dimensional
AWGN	Additive white Gaussian noise
BLER	Block error rate
CNN	Convolutional neural network
DF	Decode-and-forward
DL	Deep learning
EGC	Equal gain combining
GB	Gigabytes
GW	GateWay
IoT	Internet of Things
LoRA	Long Range
MDF	Machine learning Decode-and-Forward
ME	Equal gain combining multi-path
MM	Maximum ratio combining multi-path
mMTC	Massive machine-type communication
ML	Machine learning
MP	Multi-path
MRC	Maximum ratio combining
PA	Precision agriculture
PDD	Plant disease detection
PAN	Precision agriculture sensor network
QoS	Quality-of-service
RAM	Read access memory
SN	Sensor node
SNR	Signal-to-noise ratio
SP	Single path
TDF	Traditional decode-and-forward
TinyML	Tiny machine learning
VS	Visual sensor
WSN	Wireless sensor network

noisy data (crop image) transmitted from an onsite farm multi-path multi-hop PAN as shown in Figs. 1 and 2. This work is an extension of [11], which focused on a single-path multi-hop PAN-PDD solution. In detail, the system model is made up of (i) onsite data acquisition, transfer, and processing using autoencoder transceiver sensor nodes transferring data over multiple paths and multi-hops PAN to the gateway/central system (GW) for further processing and transfer via the internet to the offsite PDD application, and (ii) predicting whether a plant is diseased using the offsite PDD algorithm. A graphical depiction of the PAN data transfer and processing over multiple paths and multiple hops considered in this work is shown in Fig. 1. From Fig. 1, within the proposed onsite farm PAN, the crop (maize) data (leaf image) is captured using an onsite visual sensor (camera) (VS). The acquired data is pre-processed using an autoencoder algorithm (i.e., the encoder: the transmission channel effects compensation is considered in the training²) and transmitted through the farm monitoring sensor nodes (SNs) within the PAN to the PAN GW. Note that, unlike our previous work, which considered a single path for the information transfer, this work considered the data transferred using several paths within the monitoring SNs. The GW and the data receiving SNs within the multiple routing paths contain the other portion of the autoencoder (i.e., the decoder portion, which estimates the captured original plant leaf image), which processes the received plant image data. The GW/SNs then transmit the recovered data through an external network (e.g., internet + cloud) to the on/offsite PDD application to predict whether the plant is diseased. This work focused on PAN as a specific application of the proposed cascaded autoencoder nodes.³ Note that the proposed DNN PAN can be applied to other WSNs for noisy data transmission. Next, the multi-hop PAN autoencoder design is discussed.

3 Onsite farm multi-path multi-hop PAN transmission

This section discusses the details of the onsite farm PAN data acquisition, processing, and transmission to the offsite PDD application. As shown in Fig. 1, the VS captures the farm plant (leaf) image, processes the data, and transmits the data

² The compensation here refers to the consideration of the multi-hop cascading channel gains (noise) within the autoencoder training phase.

³ The PAN comprises sensor nodes (SNs) with limited computation and communication resources. Therefore, reducing the data size with an autoencoder minimizes the strain on the PAN multi-hop SNs. The data reduction and other imperfections at each node will be considered in an extension of this work.

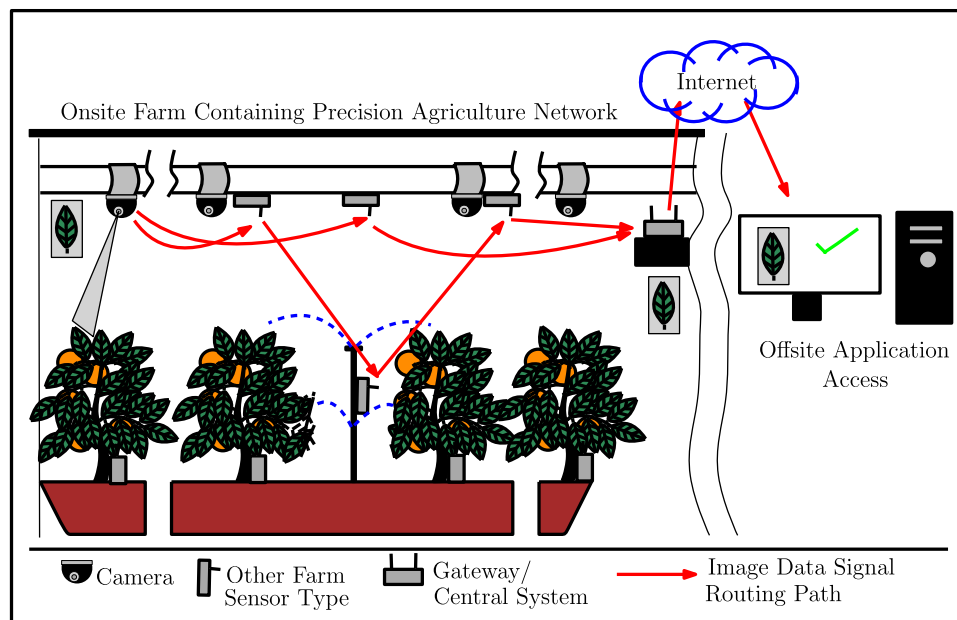


Fig. 1 On- and offsite farm deep learning multi-path multi-hop PAN architecture

through the $\sum_{i=1}^I K_i \geq 0^4$ PAN multi-path multi-hop SNs to the gateway. Each PAN SN is equipped with a single antenna. Also, the multi-hop data reception and re-transmission are done by using either a DNN (autoencoder [12, 13] which can be implemented using TinyML techniques) or the transitional DF technique [14].⁵ In this work, it is assumed that the multi-path multi-hop PAN SNs and path from the image capturing VS to the GW are already known.⁶

3.1 Signal flow for one path in the multi-path autoencoder PAN

The data processing using the autoencoder ML approach from the transmitter through the PAN multi-hop multi-path routing nodes to the onsite gateway receiver for data combining and processing is detailed below [11] and depicted in Fig. 2.

Transmitter (encoder portion) [11] At the VS, the captured image⁷ goes through the following autoencoder (encoder portion) steps before transmitting through the SNs.⁸

- Image data processing (Embedding stage): the image is converted to binary format (s) data stream (i.e., the embedding phase: conversion from 3D pixel matrix (S) to 2D decimal matrix ($\tilde{S}$) and then 1D bits vector (s)). Hence, the image data transformation is $f: S \rightarrow s$.
- DNN data processing: the s data vector is fed to the DNN for data stream dimension reduction. Here, the DNN performs the dimension reduction using a symbol mapping (i.e., number of bits per symbol (k)) to match a chosen modulation scheme (e.g., QPSK ($2^k, k = 2$), 16QAM ($2^k, k = 4$)) and PAN allowable data packet size. In detail, the image bits (s) are converted to a set of symbols $\mathcal{M}$, where each symbol consists of 2^k bits based on the modulation scheme and each symbol $\hat{S}$ is represented by a value in the set $\{1, 2, \dots, M\}$. Now, for each symbol to be transmitted, the symbol message is transformed to an $\hat{s} \in \mathbb{R}^n$ of size n , where n equals the size of transmit signal and number of data stream

⁴ Depending on the selected optimal multi-hop multi-path and VS location; $K_i = 0$ or $K_i \geq 1$ for VS-to-GW direct or i multi-paths with K_i multi-hop transmission, respectively.

⁵ The DF technique involves each relaying node decoding its received data signal before re-transmitting the decoded data to the next relaying SN. TinyML is a type of ML implementation that allows models to run on smaller, low-power devices such as PAN sensors.

⁶ Future work will consider the development of an efficient routing algorithm.

⁷ In this proof of concept work, leaf images from PlantVillage were used [15].

⁸ The autoencoder in this work was based on the modification of the wireless communication autoencoder model proposed and used in [12, 13, 16]. Also, it is the same as the one presented in our previous work in [11].

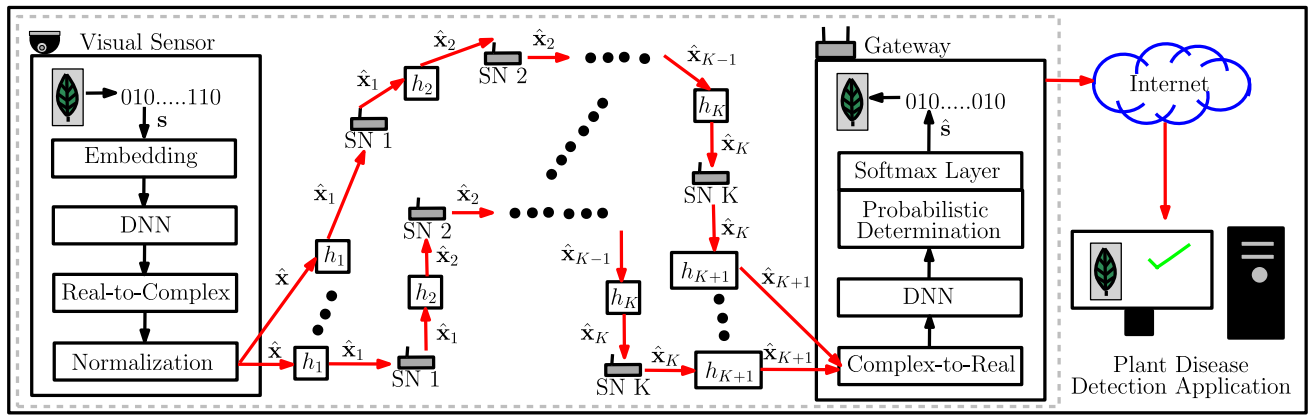


Fig. 2 Autoencoder representation for a multi-path multi-hop PAN communication model

$\hat{s}$ accessed channels. In summary, the encoder portion does the data modulation based on learned channel state $f: \mathbf{S} \rightarrow \hat{\mathbf{S}} \rightarrow \mathbf{s} \rightarrow \hat{\mathbf{S}} \in \{1, 2, \dots, M\} \rightarrow \hat{\mathbf{s}} \in \mathbb{R}^n$.

- c. Transmit data normalization: the resulting data stream is converted to a complex-valued data set (i.e., equal portions of real and imaginary components). Finally, the transmit data is normalized to satisfy the transmit power constraint, that is, $\hat{\mathbf{x}}_0 = \|\hat{\mathbf{s}}\|^2 \leq P_{VS}$.

PAN multi-hop The VS data stream is transmitted through G_i SNs per path (i) to the GW for final decoding and transmission to the offsite unit. This implies that the total routing paths (I) and SNs are totaled as $\sum_{i=1}^I G_i$. The data received at a multi-hop SN is deduced as

$$\hat{\mathbf{x}}_{g_i} = h_{g_i} \hat{\mathbf{x}}_{g_i-1} + \phi_{g_i}, g = 1, 2, \dots, G_i, \quad (1)$$

where h_{g_i} represents the single-path inter-node channel (modeled as additive white Gaussian noise (AWGN) channels) and ϕ_{g_i} represents the antenna noise at single-path SN

g_i . Each SN uses the MDF (autoencoder) to decode and re-transmit information. The received signal at the GW from a single path is given as

$$\hat{\mathbf{x}}_{G_i+1} = h_{G_i+1} \hat{\mathbf{x}}_{G_i} + \phi_{G_i+1}. \quad (2)$$

Receiver (GW) [11] At the GW, the decoder portion of the autoencoder is implemented as follows.

- The received complex single-path data is concatenated (real and imaginary components, $\hat{\mathbf{x}}_{G_i+1} \rightarrow \tilde{\mathbf{x}}_i$), and the real value components are passed to the DNN after the multiple path signals are combined.
- At the GW, all the multiple path signals received are combined using equal gain combining and maximum ratio combining, which are represented respectively as

$$\tilde{\mathbf{x}} = \frac{1}{I} \sum_{i=1}^I \tilde{\mathbf{x}}_i, \text{ and } \mathbf{\tilde{x}} = \sum_{i=1}^I \frac{h_{G_i}}{\sum_{i=1}^I |h_{G_i}|^2} \tilde{\mathbf{x}}_i. \quad (3)$$

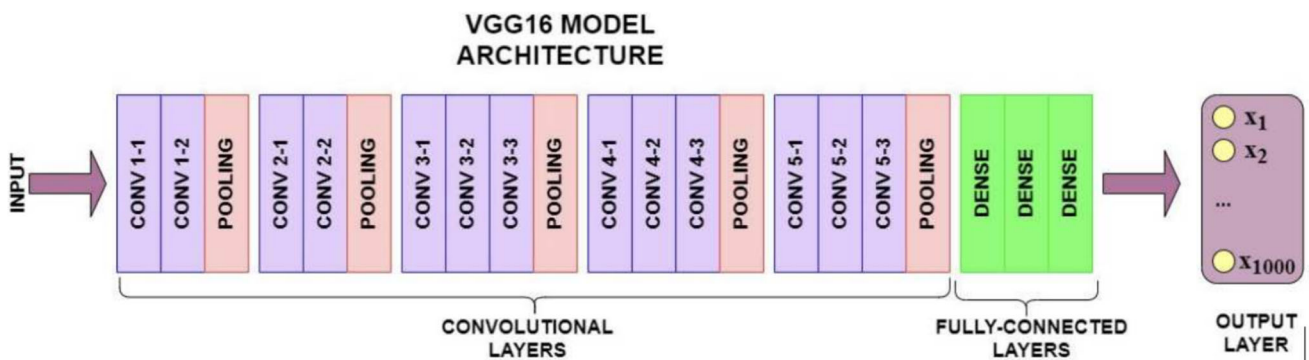


Fig. 3 Architecture of the VGG16 model [17]

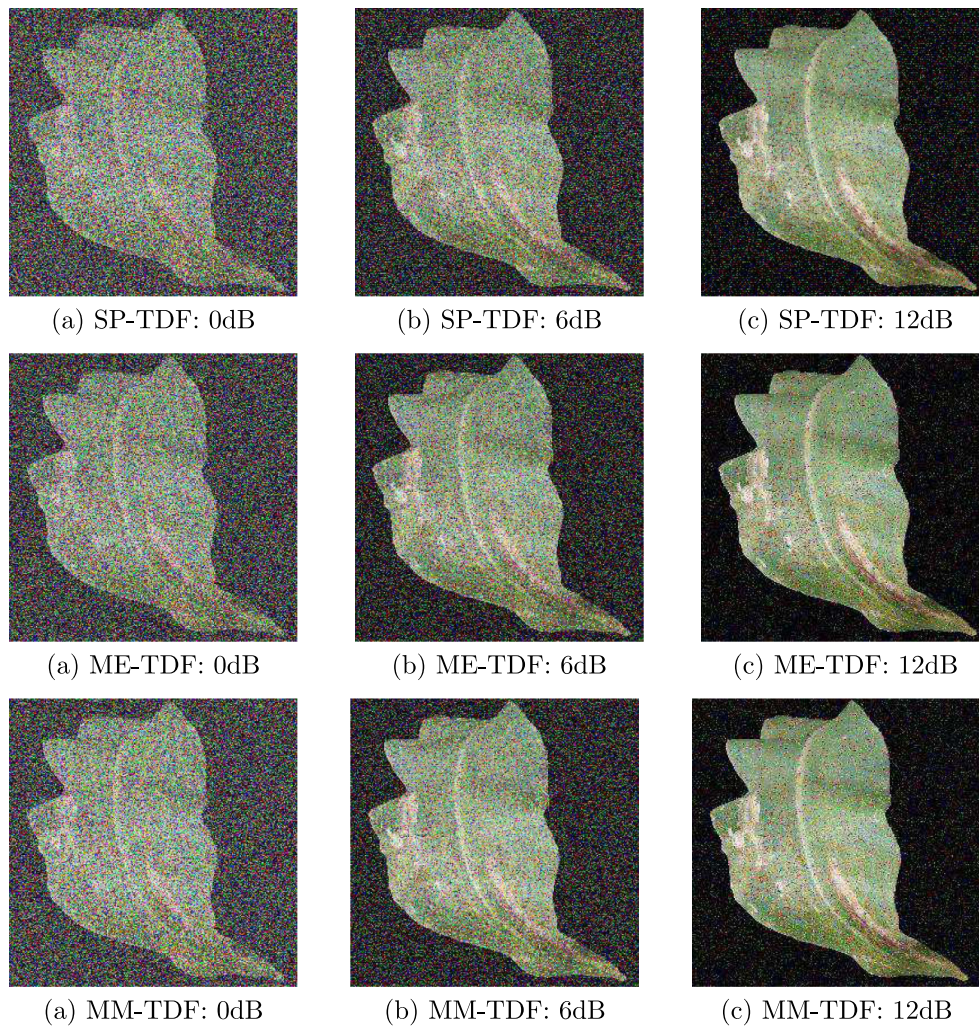
Table 2 DNN multi-path multi-hop PAN simulation setup

Parameter	Value
Inter-node SNR E_0	0 to 12 dB
Channel model	AWGN
{paths, relays, bits, channels} number	$\{I = 2, K_i = 2, b = 2, n = 2\}$
Number of plant images	100
Plant disease	Maize common rust

c. DNN data processing: the reverse of the DNN in the encoder occurs here, where the n data stream vector is transformed to an estimated symbol of size 2^k bits using softmax function (softmax layer) ($\tilde{\mathbf{x}} \rightarrow \tilde{\mathbf{s}}$).

d. Image recovery: the estimated 1D image bit vector $\tilde{\mathbf{s}}$ is reshaped to the estimated 2D decimal matrix ($\tilde{\tilde{\mathbf{S}}}$), then to the estimated 3D pixel matrix ($\tilde{\mathbf{S}}$). The total decoding process is presented as $f : \tilde{\mathbf{x}} \in \mathbb{R}^n \rightarrow \tilde{\mathbf{S}} \in \{1, 2, \dots, M\} \rightarrow \tilde{\mathbf{s}} \rightarrow \tilde{\tilde{\mathbf{S}}} \rightarrow \tilde{\mathbf{S}}$.

The recovered image data is then transmitted over the internet to the DNN PDD offsite system for disease detection predictions. Note, the training of the autoencoder involves the use of random generated bits as input and outputs for the training and validation sets of the autoencoder [12, 13]. In addition, the autoencoder training and validation incorporated the channel characteristics into the learning process.

**Fig. 4** Example of the TDF recovered images for increasing signal-to-noise transmit power values

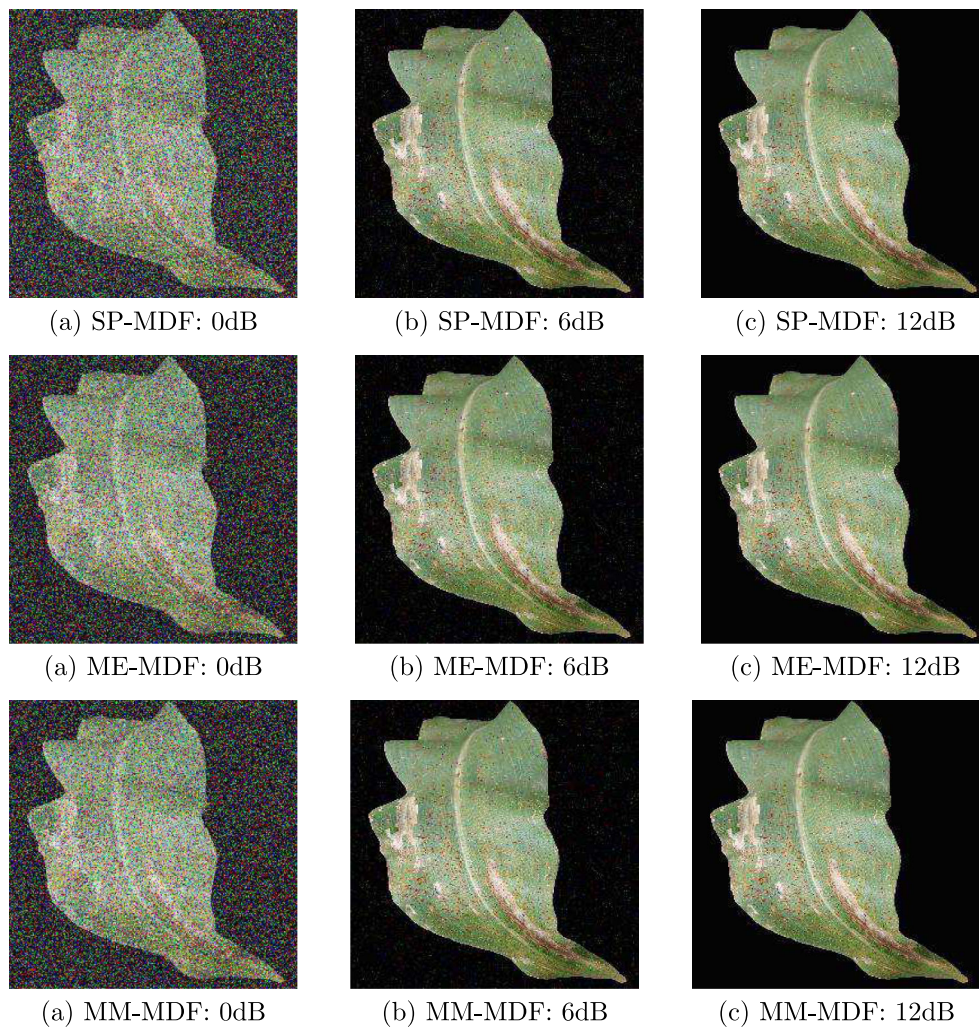


Fig. 5 Example of the MDF recovered images for increasing signal-to-noise transmit power values

3.2 Signal flow for multi-path multi-hop traditional decode-and-forward PAN (benchmark)

A summary of the signal flow for traditional DF multi-path multi-hop routing PAN used in this work is presented as follows [11].⁹

Transmitter The image data processing portion for the DF technique is the same as that of the encoder approach. Similarly, the 1D bits data ($\mathbf{s}$) is then modulated using the chosen modulation scheme ($\hat{\mathbf{s}}$) and transmitted over several/single

packets depending on the device coherence time allowable packet size. Note, $\hat{\mathbf{x}}_0 = \|\hat{\mathbf{s}}\|^2 \leq P_{VS}$ must be satisfied.

PAN multi-path multi-hop Transmission through the $\sum_{i=1}^I G_i$ SNs is achieved through the traditional DF (TDF) approach, where the received data (Eq. 1) as SN g_i is decoded and then decoded information is re-transmitted (Eq. 2) to SN g_i or GW.

Receiver (GW) At the GW, the received data is combined as presented in (3), demodulated, and the image data is recovered using the same encoder image recovery process. Next, the DNN PDD is presented.

⁹ An argument can be made for the VS (Visual sensor) having the capability to run the PDD algorithm (via TinyML) and instead forward the prediction data. However, this may require additional hardware and programming embedded into the VS devices. This will increase the purchase cost of devices and operational expenses (VS node battery consumption, replacement, and lifespan), proportional to the number of VS devices needed for a farm. Also, this defeats the purpose of using mMTC devices for smart precision farming equipped with 5G and beyond technologies.

4 Plant disease detection using CNN

The acquired PAN image data must be used in PDD training to test and affirm our noisy data theory and our proposed cascaded encoder design for PAN-PDD. Hence, we briefly

discuss the pre-existing PDD algorithm used to confirm our deductions and proposed solution.

The high-precision and widely used VGG16 is adopted in this work for the PDD based on the images acquired from the PAN simulation. The architecture of the VGG16 is depicted in Fig. 3. The VGG16 consists of the typical CNN model with three layers: a convolutional layer, a pooling layer, and a fully connected layer [18]. The VGG16 is adopted because of its high precision in PDD, as shown in [19]. We pre-train on the PlantVillage datasets. Further details on the ML implementation are provided in Section 5.1.1.

5 System evaluation

5.1 Onsite farm multi-path multi-hop PAN simulation

5.1.1 Simulation setup

The multi-path multi-hop PAN simulation setup is summarized in Table 2. Details of the autoencoder implementation adopted for the data routing within a single path follow a similar structure as those found in [12, 13, 16]. The plant (corn) images used for the PAN and DNN PDD simulations were acquired from “PlantVillage” [15]. We acquired 1600 unhealthy and 1600 healthy noisy plant images from the PAN image transmission simulations. Two paths (i.e., $I = 2$) are used within the simulation for the multi-paths, and two routing (i.e., $K_i = 2$) hops per each routing path. Next, the results (i.e., acquired noisy images) from the PAN simulation are presented. The Block Error Rate (BLER) is chosen as the quality-of-service (QoS) to discuss data quality. For comparison, the MDF is categorized into the single-path multi-hop (SP-MDF [11]), equal gain combining multi-path multi-hop (ME-MDF), and the maximum ratio combining multi-path multi-hop (MM-MDF) schemes. Similarly, the TDF consists of the single-path multi-hop (SP-TDF [11]), equal gain combining multi-path multi-hop (ME-TDF), and the maximum ratio combining multi-path multi-hop (MM-TDF) schemes. In addition to the single-path benchmarks, two upper-bound benchmarks, namely, the single-path (SP-CDF) and multi-path (ME-CDF and MM-CDF) coded DF, are presented in the BLER plots. This is to evaluate the performance of the MDF schemes and affirm the simulations are accurate. Each BLER plot was acquired using 10^3 iterations (experiments).

5.1.2 Simulation discussion

Figures 4 and 5 show the “recovered” plant images at the GW for both the TDF (i.e., SP-, ME-, and MM-TDFs, respectively) and MDF (i.e., SP-, ME-, and MM-MDFs, respectively) approaches for SN transmit powers of 0 dB,

6 dB, and 12 dB. It can be seen from the recovered images that the MDF approach has better image quality than the TDF approach. This is because the MDF approach is based on a trained ML that considers the channel effect. However, this does not apply to the TDF approach, which is uncoded and does not consider the channel effect. Even though both schemes have the worst images at 0 dB, the TDF scheme is worse than the MDF scheme. The ME- and MM- considerations for the MDF and that of the TDF have the same image quality. This implies that the equal gain combining and maximum ratio combining result in a similar BLER. Also, the TDF schemes exhibit image distortions amongst all three considered SNR values. The MDF schemes see near-perfect images for SNR greater than 6 dB.

The effects of the noisy data decoding are presented in Fig. 6. Generally, as the SNR increases, the BLER for all schemes improves. From Fig. 6, it is observed that there is a large BLER performance gap between the {MDF + CDF} schemes compared to the TDF schemes. In addition, there are slight performance differences between the various subschemes (SP-, ME-, and MM-) for the TDF scheme. However, this is not the same behavior for the CDF and MDF schemes. Even though the CDF outperforms the MDF schemes marginally, both schemes’ ME- and MM-subschemas slightly outperform their SP- subscheme counterparts, respectively.

Next, the influence of increasing the number of routing hops (multi-hops) has on the BLER in Fig. 9. It is observed from the figure that as the number of hops increases, the BLER worsens (increasing value). This observation is because as the number of routing hops increases, the amount of channel effect (data noisy effect) increases, increasing the BLER. As expected, the MDF and the CDF schemes outper-

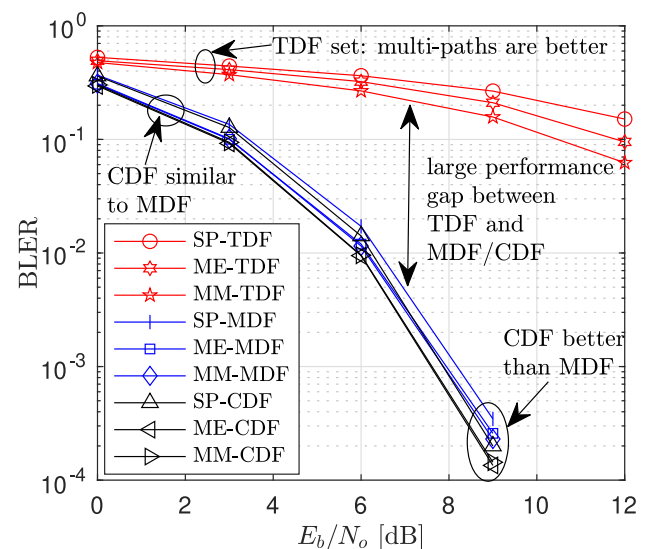


Fig. 6 PAN BLER vs SNR with $K_i = 2$

form the TDF approach. The performance gap between SP-, ME-, and MM-TDF remains constant for increasing K_i . The ME- and MM-MDF have the same performance. This behavior is the same for the ME- and MM-CDF. However, there is a constant gap between the multi-path MDF and CDF schemes and their respective single-path approach. The BLER for the CDF, MDF, and TDF is approximately between the range 10^{-3} and 10^{-2} .

This influence of noisy data is observed in the sample image data set for increasing routing hops (multi-hop (K_i)) are presented in Figs. 7 and 8 for TDF and MDF, respectively. Here, the TDF and MDF approaches maintain similar image quality with increasing $K_i = \{2, 6, 10\}$ because of the small recorded BLER range of 10^{-3} to 10^{-2} at an SNR value of 8 dB. However, the TDF approaches have worse images compared to the MDF approaches (Fig. 9).

Our simulation results on BLER clearly demonstrate the pressing need for improved data transfer quality in precision farming communication networks. It can easily be observed

that the images acquired for the proposed solutions are far clearer compared to the traditional solutions. Additionally, the slight differences in BLER, with the proposed scheme having superior values, show that the proposed solution has a better quality of service. The potential of machine learning in communication to address the issue of noisy data transfer by mitigating the noisy channel effect is significant. By implementing an SN autoencoder within PAN, we can immediately begin to improve the data transfer link performance and ensure the improved future success of PDD in on-field precision farming through noisy data transfer improvement by machine learning techniques.

5.2 Plant disease detection

5.2.1 Dataset

A sample dataset from PlantVillage [15, 20] is extracted and used as discussed above to assess the impact of PAN on the

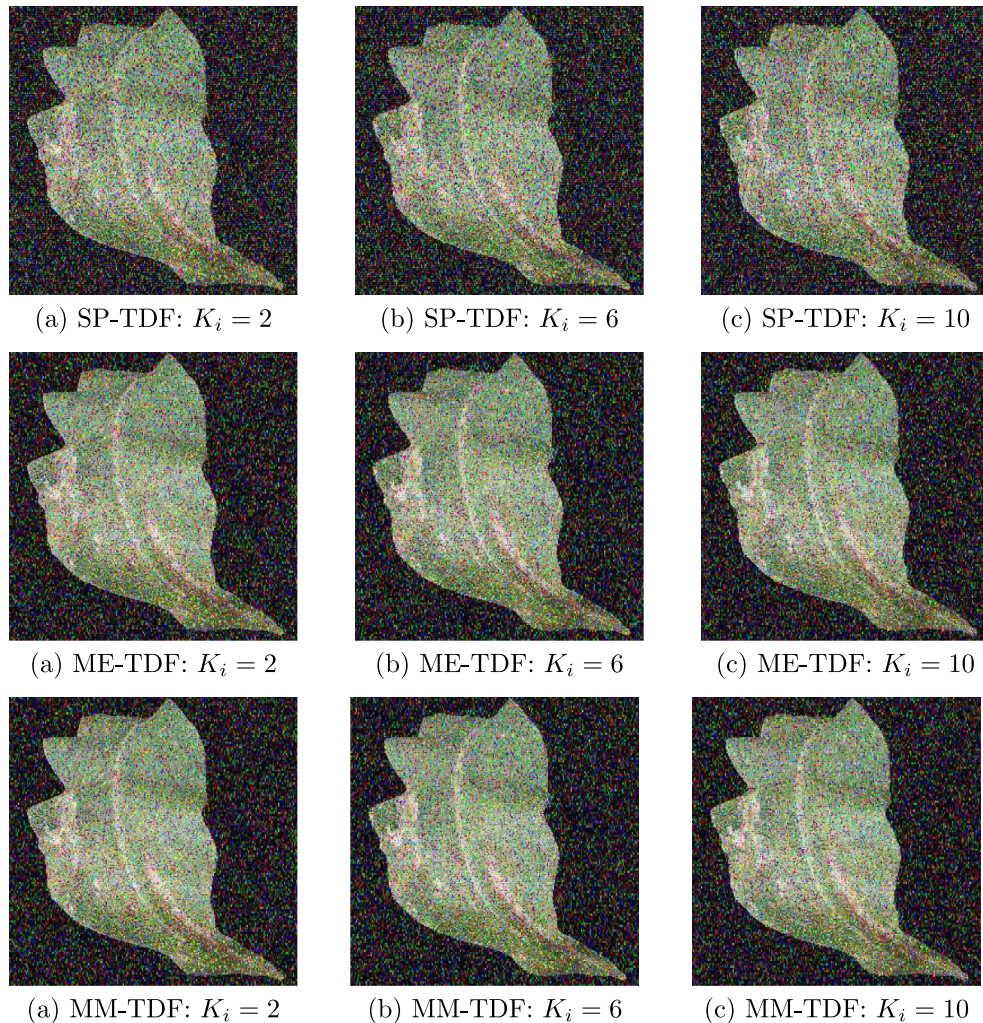


Fig. 7 Example of the TDF recovered images for increasing number of routing hops

performance of PDD models. This implies that the PlantVillage images are simulated to pass through the PAN network to the gateway, acquiring the noisy image data for the PDD prediction. The Corn dataset contains plant images (i.e., corn healthy and common rust unhealthy image datasets) essential for training an effective PDD model. The PlantVillage dataset was chosen because it is the most used open-source plant disease dataset. The sizes and distribution of the datasets are found in Table 3.

5.2.2 Simulation setup

The DNN PDD experiments were run on an Apple MacBook computer with an M1 Max processor and 32GB RAM. Microsoft Visual Studio [21] operating Python 3.9 was used to run the model. For each dataset, an {80%, 20%} split is made where the 20% is used for testing, and the 80% is further split into 80% training and 20% validation sets. We used

two transfer learning models (i.e., VGG16 model) from Keras [22] for the DNN PDD development and proof of concept of PDD training based on noisy image data. These models were utilized to classify the images. We used transfer learning to fine-tune both models for plant disease detection using the PlantVillage dataset. The fully connected layers of the model were removed, and two layers were frozen during the training, switching from trainable to non-trainable. Table 4 shows the hyperparameters used for the experiments.

5.2.3 Simulation discussion

Results and discussion presented on the PDD test results are presented in Table 5 and are based on the following metrics:

- i **Accuracy:** It represents the overall performance of the DNN PDD algorithm in ability to predict the right classification (i.e., healthy and unhealthy crops), which is

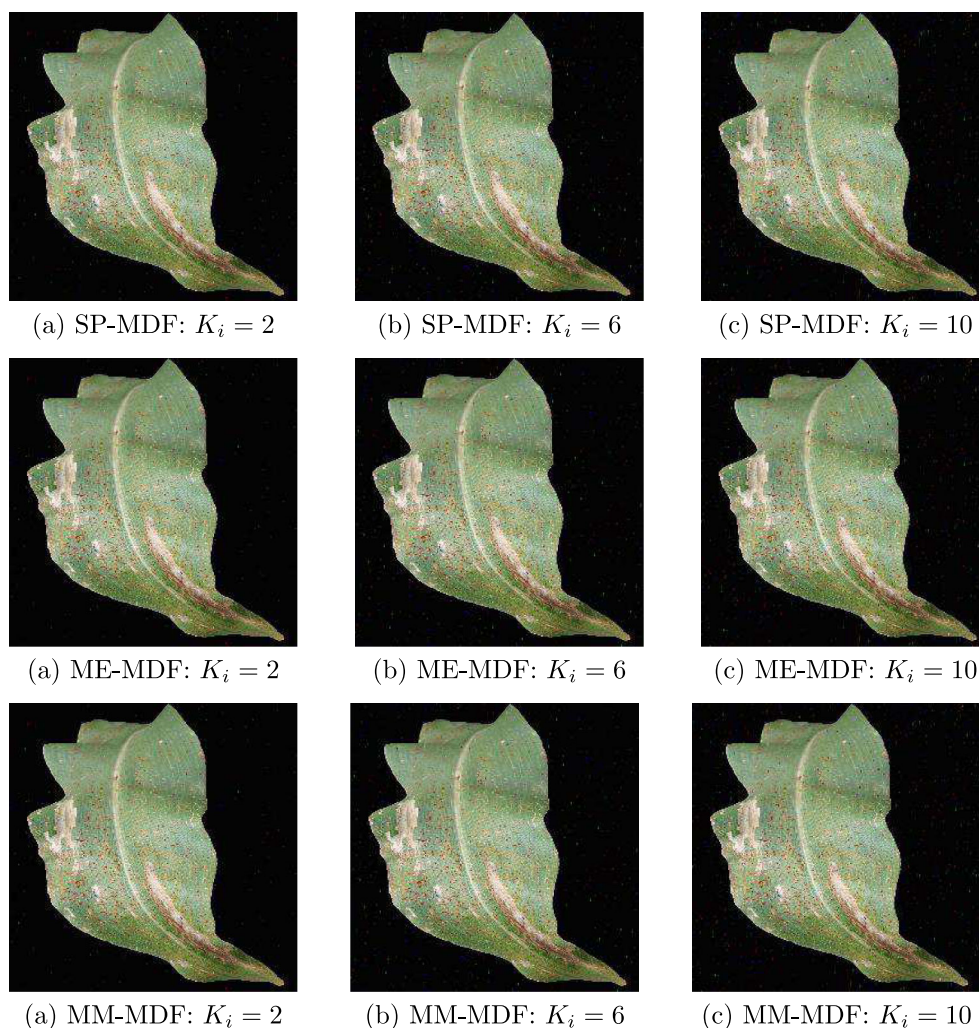


Fig. 8 Example of the MDF recovered images for increasing number of routing hops

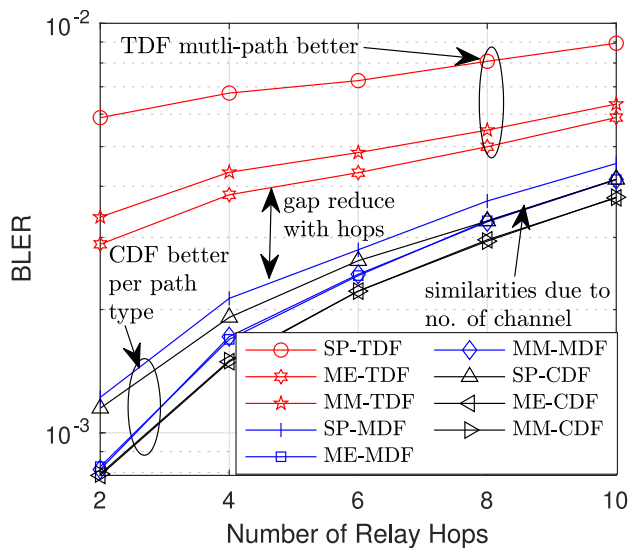


Fig. 9 PAN BLER vs K_i with $E_b/N_o = 8\text{dB}$

mathematically represented as

$$\text{accuracy} = \frac{\text{true healthy} + \text{true unhealthy}}{\text{true healthy} + \text{true unhealthy} + \text{false healthy} + \text{false unhealthy}}. \quad (4)$$

- ii **Precision:** It represents the overall performance of the DNN PDD algorithm in its ability to truly determine the unhealthy crops out of all the positive predictions made. The precision is determined mathematically as

$$\text{precision} = \frac{\text{true unhealthy}}{\text{true unhealthy} + \text{false unhealthy}}. \quad (5)$$

- iii **Recall:** It represents the performance capability of the DNN PDD algorithm to determine all unhealthy crops, that is, out of all the data predictions that should be unhealthy crops, how many were predicted correctly. The recall is deduced analytically as

$$\text{recall} = \frac{\text{true unhealthy}}{\text{true unhealthy} + \text{false healthy}}. \quad (6)$$

Table 3 Overview of the training and testing datasets

Dataset	Healthy	Diseased
Corn (default, TDF, MDF)	(800, 800, 800)	(800, 800, 800)
Total	2400	2400

Table 4 Overview of VGG16 and Inception V3 hyperparameters

Hyperparameters	VGG16
Input size	(224, 224, 3)
(Batch size, learning rate)	(32, 0.0002)
(Number of epochs, optimizer)	(15, Adam)

- iv **F1-score:** It represents the performance ratio between the precision and recall. It is defined mathematically as

$$F1 = 2 \left(\frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \right). \quad (7)$$

The results show that both transfer learning VGG16 models can distinguish the diseased images from the healthy images for all three datasets. Also, the DNN PDD performance for the MDL (i.e., SP-, ME-, and MM-MDFs) images is marginally better than that of the TDF images (i.e., SP-, ME-, and MM-TDF) because the images were augmented (perfectly cropped) and high-definition images from PlantVillage. However, on-field real-time captured images are expected to be of low definition (i.e., low quality and blurry/distorted captured images due to low equality and specification cameras) and to have blockages such as other leaves, branches, and other plant parts, impeding the desired plant leaf area of focus present in the captured images. These camera usage abstractions and distortions will be further increased because of the data transmission noise. This leads to a reduced ability to isolate the plant leaf area of focus for DNN PDD training and prediction, unlike the perfectly isolated high-definition images used in this work. Hence, the performance differences between the usage of on-field low-definition captured images and the perfectly isolated high-definition images used in this work will be significant. This effect or prediction is affirmed from the results presented. Comparing the results of perfectly isolated high-definition images passed through the noisy multi-hop

Table 5 VGG16 results on PAN generated datasets

Dataset	Accuracy	Precision	Recall	F1-score
Default	100.000%	100.000%	100.000%	100.000%
SP-TDF	96.000%	99.291%	93.333%	96.220%
ME-TDF	97.665%	98.462%	96.970%	97.710%
MM-TDF	95.720%	99.187%	92.424%	95.686%
SP-MDF	97.276%	98.450%	96.212%	97.318%
ME-MDF	98.054%	98.473%	97.727%	98.099%
MM-MDF	97.276%	96.992%	97.727%	97.358%

(i.e., both single- and multi-path) relay network to results acquired from the perfect images from PlantVillage, trained directly without going through the PAN simulation, had perfect results (100%). Note that the default had perfect results because the ideal (original) images, which are not practical images, were used.

6 Conclusion

This paper presented work on onsite farm multi-path multi-hop PAN using traditional (TDF) and ML (MDF) decode-and-forward approaches and farm offsite deep neural network (DNN) plant disease detection (PDD) implementation. It was shown that the MDF approach is better compared to the TDF approach, and image acquisition for DNN PPD offsite is better. However, for the DNN PDD execution, the MDF and TDF had similar prediction values. The following research stage involves designing and incorporating both a spectrum- and energy-efficient routing protocol based on traditional and ML approaches. In addition, other modulation schemes will be considered for the system model. Finally, physical implementation will be done.

Open access

This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Acknowledgements The research leading to the results reported in this work received co-funding from the European Union's Horizon 2020 research and innovation program under the Marie Skłodowska-Curie grant No. 8995462, the Region of Bretagne, France, and IMT-Atlantique, Brest, France. This agreement is signed between the Region of Bretagne and IMT-Atlantique, Brest, France.

Author contribution All authors reviewed and contributed to developing the research idea and the manuscript writing and review. K. E. Bennin also worked on plant disease detection (PDD) coding, simulation, and results presentation. K. E. Bennin also worked on the PDD tables, figures and discussions. K. E. Bennin also worked on the editing of the manuscript. D. K. P. Asiedu also worked on the precision agriculture network (PAN) and PDD coding, simulation, and results presentation. D. K. P. Asiedu also worked on the figures within the

manuscript and the presentation and discussion of the PAN tables and plots. D. K. P. Asiedu worked on the writing of the whole manuscript. D. A. N. Gookyi worked on the manuscript editing and the PDD coding. M. Benjillali worked on the communication model development, coding, and the editing of the manuscript. S. Soaudi worked on the communication model development, coding, and the editing of the manuscript.

Funding Open access funding is provided by IMT-Atlantique within the Marie Skłodowska-Curie grant agreement No. 8995462.

Data availability No datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare no competing interests.

References

- Polymeni S, Plastras S, Skoutas DN, Kormentzas G, Skianis C (2023) The impact of 6G-IoT technologies on the development of agriculture 5.0: A review. *Electronics* 12(12):2651
- Tang Y, Dananjayan S, Hou C, Guo Q, Luo S, He Y (2021) A survey on the 5G network and its impact on agriculture: challenges and opportunities. *Comput Electron Agric* 180
- Shehzad MK, Rose L, Butt MM, Kovács IZ, Assaad M, Guizani M (2022) Artificial intelligence for 6G networks: technology advancement and standardization. *IEEE Veh Technol Mag* 17(3):16–25
- Mohanty SP, Hughes DP, Salathe M (2016) Using deep learning for image-based plant disease detection. *Frontiers Plant Sci* 7:1419
- Khan MA, Khan A, Abuibaid M, Huang JS (2023) Harnessing 5G networks for enhanced precision agriculture: challenges and potential solutions. In: *Proc Int Conf Smart Applications, Commun Network*, pp 1–6
- Roberts MJ (2006) The value of plant disease early-warning systems: a case study of USDA's soybean rust coordinated framework vol 18. USDA Economic Research Service, ???
- Abbas A, Jain S, Gour M, Vankudothu S (2021) Tomato plant disease detection using transfer learning with C-GAN synthetic images. *Comput Electron Agric* 187:106279
- Min B, Kim T, Shin D (2023) Shin D Data augmentation method for plant leaf disease recognition. *Appl Sci* 13(3):1465
- Ahmad M, Abdullah M, Moon H, Han D (2021) Plant disease detection in imbalanced datasets using efficient convolutional neural networks with stepwise transfer learning. *IEEE Access* 9:140565–140580
- Picon A, Alvarez-Gila A, Seitz M, Ortiz-Barredo A, Echazarra J (2019) Johannes A Deep convolutional neural networks for mobile capture device-based crop disease classification in the wild. *Comput Electron Agric* 161:280–290
- Asiedu DKP, Ofori-Amanfo KB, Bennin KE, Benjillali M, Lee K-J, Gookyi DAN, Saoudi S (2024) Precision agriculture deep neural network driven multi-hop plant image noisy data transmission and plant disease detection. In: *Proc ISIVC 2024: IEEE International Conference on Signal, Image, Video and Communications*
- Dörner S, Cammerer S, Hoydis J, Ten Brink S (2017) Deep learning based communication over the air. *IEEE J Sel Topics Signal Process* 12(1):132–143
- O'shea T, Hoydis J (2017) An introduction to deep learning for the physical layer. *IEEE Trans Cog Commun Netw* 3(4):563–575
- Asiedu DKP, Lee H, Lee K-J (2019) Simultaneous wireless information and power transfer for decode-and-forward multihop relay

- systems in energy-constrained IoT networks. *IEEE Int Things J* 6(6):9413–9426
15. Village P (2023) Data for: Identification of plant leaf diseases using a 9-layer deep convolutional neural network. PlantVillage, Date Accessed. <https://data.mendeley.com/datasets/tywbtsjrjv/1>
 16. R2023a M (2023) Matlab documentation: autoencoders for wireless communications. MathWorks, Date Accessed. <https://fr.mathworks.com/help/comm/ug/autoencoders-for-wireless-communications.html>
 17. McDermott J (2021) Hands-on transfer learning with Keras and the VGG16 model
 18. Lu J, Tan L, Jiang H (2021) Review on convolutional neural network (CNN) applied to plant leaf disease classification. *Agriculture* 11(8):707
 19. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*
 20. Hughes D, Salathe M (2015) An open access repository of images on plant health to enable the development of mobile disease diagnostics. *arXiv preprint arXiv:1511.08060*
 21. Visio M (2023) Visual studio code: code editing redefined. PlantVillage, Date Accessed. <https://code.visualstudio.com/>
 22. Visio M (2023) Keras: simple flexible powerful. PlantVillage, Date Accessed. <https://keras.io/>

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.

Authors and Affiliations

Derek K. P. Asiedu¹ · Kwabena E. Bennin² · Dennis A. N. Gookyi³ · Mustapha Benjillali^{1,4} · Samir Saoudi¹

✉ Derek K. P. Asiedu
derek.asiedu@imt-atlantique.fr

Kwabena E. Bennin
kwabena.bennin@wur.nl

Dennis A. N. Gookyi
dennisgookyi@gmail.com

Mustapha Benjillali
benjillali@ieee.org

Samir Saoudi
samir.saoudi@imt-atlantique.fr

¹ Department of Mathematical and Electrical Engineering, IMT-Atlantique, Lab-STICC, 655 Av. du Technopôle, Brest 29238, Brittany, France

² Information Technology Group, Wageningen University and Research, 10587 Wageningen, Netherlands

³ Institute for Scientific and Technological Information, The Council for Scientific and Industrial Research, Agostinho Neto Road, Council Close, Accra 610101, Greater-Accra, Ghana

⁴ Communication Systems Department, Institut national des postes et télécommunications, 10100 Rabat, Morocco