Original Article

Hybrid Energy Harvesting Model for Attaining Energy Neutrality in IoT-based Smart Agricultural System

Rakshith Nagaraj[1], Minavathi[2]

[1,2]*Department of Computer Science and Engineering, P.E.S. College of Engineering, India*

[1]*Corresponding Author : rakshithn91@gmail.com*

Abstract - *Artificial Intelligence (AI) has constantly entered various applications in economical ways, specifically in control and monitoring applications in the sector related to agriculture. Moreover, the process of attaining neutrality in the energy for the Internet of Things (IoT) remains a challenging task. The agricultural sector is still affected by various pest diseases, which lead to danger to the productivity of crops and economically affect the farmers. To overcome these challenges, this research introduced an energy harvesting system to achieve the state of energy neutrality in IoT-based smart agricultural systems. Moreover, the Maximum Power Point Tracking (MPPT) algorithm is utilized to attain the state of energy neutrality, and the Principal Component Analysis (PCA) is used to detect the pests for the provided Region of Interest (ROI). The Xception model is utilized to classify the codling moths and general insects affecting the crops. The Raspberry Pi3 (RPi3) is utilized to collect images in a single board computer and helps detect the pests accurately. The hybrid energy harvesting system is a combination of solar, wind and grid harvesters. Li-Po battery with 1820 mAh is utilized to charge the Hybrid system using the Xception model. The experimental results show that the proposed hybrid energy harvesting system consumed minimum energy of 118.2 J while the existing methods, such as Smart Energy Harvesting using Wireless Sensor Networks (SEH-WSN) and Long Range – Low Power Wide Area Networks (LR-LPWAN), consumed energy of 129.8 J and 124.5 J respectively.*

Keywords - Energy neutrality, Hybrid energy harvesting system, Internet of things, Raspberry pi, Smart agriculture.

1. Introduction

The increase in the world's population leads to an increase in the demand for food production. According to the report of the Food and Agriculture Organization (FAO), it is estimated that the world population will reach around 9.1 billion in the year 2050. To provide sufficient food for the population, food productivity must be improved by 70% [1,2]. Moreover, the factors such as crop diseases, drought, and weeds may lead to a severe drop in food productivity [3]. Therefore, precise and periodic monitoring of the plants' health conditions has become a significant process [4]. The development of sensor technologies enabled IoT applications and helped to improvise the quality and quantity of agricultural production with cost reduction [5]. IoT technology has paved the way for developing various techniques such as Wireless Sensor Networks (WSN), edge computing, and other web services applicable for various farming applications such as irrigation, growth monitoring, fertilization, etc. [6]. Generally, the IoT sensors consist of AC grids where complexity occurs in their construction and lacks in providing long-term energy supply. This energy issue has limited the construction of IoT networks in the sensor networks related to agriculture [7].

More than that, providing electrical supply to the Wireless Sensor Networks (WSN) and IoT in remote areas is quite challenging to develop smart agriculture techniques [8]. In ordinary natural farming, farmers need to visit the agricultural fields to compute the condition of crops. On average, farmers need to spend around 70% of their time evaluating and understanding the crop's condition [9]. The emergence of pests and illnesses during crop development is intimately tied to climate change. IoT technology collects information about entities [10,26] and allows for an easy way to monitor crop growth activities in real-time. However, most farms are generally located in remote areas with limited access to infrastructure, limiting the development and use of agricultural IoT technologies [12,13]. The sensors equipped with the nodes of IoT gather the relevant data from the environment and help to maintain the balanced agricultural ecosystem. Alternate energy sources are necessary for sustainable smart agricultural farming to create a high-intensity environment without creating an energy demand [14,15].

The major contributions of this research are listed as follows:

1. The improvisation and description of smart traps related to IoT have been done in the absence of human intervention.
2. The Xception model is trained and optimized to detect and classify the codling moths and general insects.
3. The hybrid energy harvesting system is utilized to charge the Li-Po battery with an 1820 mAh battery. The hybrid energy harvesting system combines solar, wind, and grid harvesters.

2. Research Gap

The limitations observed from the existing research are high energy consumption, low consideration of climatic conditions for real-time energy harvesting applications, interface issues, more processing time and computational time, high-cost maintenance, and time delay. To overcome this limitation, an energy harvesting system is proposed to achieve the state of energy neutrality in IoT-based smart agricultural systems. An MPPT algorithm is employed in the system to neutralize the energy received from the solar panels and transmit it to the harvesting system. The PCA is used to detect the pests in the ROI. This experiment's output results constitute the development of a smart agriculture system and better detection of pests.

The rest of the paper is structured as follows: Section 2 is discussed the related works. The proposed method is presented in Section 3. The results and analysis of the proposed method are discussed in Section 4; finally, the paper's conclusion is presented in Section 5.

3. Related Works

Himanshu Sharma [16] has introduced a framework to provide an enhanced life for the applications related to Smart Energy Harvesting using Wireless Sensor Networks (SEH-WSN). The introduced framework improves the life span of WSN networks using the SHE method. The SHE-WSN framework consists of a battery charging circuit and solar panels attached to the nodes of WSN. The nodes in WSN sense the activities based on the duty cycle, and this sensed data is transferred to the sink node. These sink nodes perform communication directly with the sensors and help in agricultural activities. The framework increases the network throughput and improvises the lifespan of WSN. However, the energy consumption of SEH – WSN nodes are higher in the absence of optimization algorithms.

Murtaza Cicioglu and Ali Çalhan [17] have introduced a sensitive agriculture-based integrated system on WSN and drone communication. The sensors with various packets were created, and these packets were transmitted to the coordinator node. The drone was utilized to gather the information of appropriate packets from the coordinating nodes, and the transmission of packets takes place at the gateway to reach the target. The integrated system provides better results with less delay and throughput, which is valid for applications related to smart farming. However, the integrated system does not consider the geographical conditions and climatic effects while accessing smart farming applications.

Weidang Lu [18] has introduced an architectural design of smart agriculture using Simultaneous Wireless Information and Power Transfer (SWIPT). The communication of SWIPT with the WSN was categorized into two stages; in the first stage, the information was transferred from the source sensors to the relay sensors and destination sensors. The relay sensors utilize a specific part of the subcarriers to gather the information, while the destination sensors utilize all the subcarriers to gather the information. The system achieved maximum energy efficiency by using the transmitted information from the optimized power allocation to transmit energy and information. Due to the process with the single sensor node, interference problems occurred during the transmitting and receiving of the information.

Himanshu Agarwal [27] has introduced a product density model for the IoT-enabled base station to provide precision agriculture. Moreover, the model utilized an improved duty cycling algorithm to attain better energy-neutral operations and improvise the network's life span. The improved duty cycling algorithm was utilized to select an optimal path for data transmission and helped to discover the possible pathways from the source node to the base station. The introduced product density model could model the time–dependent random variations. However, the product density model was not suited when it was tested with the networks based on physical IoT.

Tran Anh Khoa [20] has introduced a topology with the sensor nodes to monitor the level of water and moisture content present in the soil and rain prediction sensors using Long Range – Low Power Wide Area Network (LR-LPWAN). The circuit board containing LoRa-LPWAN was optimized by conjoining the layers and software implementation. The system detects the measured values and intimate the users through a network or mobile applications. The LoRa-LPWAN can monitor two or more agricultural fields with the same mobile app, which has varying growth schedules. However, the system stores and computes the data to detect an optimal solution.

Zhixin Wang [21] has introduced a Hybrid Energy Harvesting Device (HEHD), which was utilized in converting wind and solar energy into electrical energy to provide a self-powered smart agricultural system. The HEHD was a combination of electromagnetic generators, triboelectric Nanogenerators and solar cells to collect energy. In HEHD, the rotational motion was transformed into translational motion using a transverse connector. The

HEHD method helped in self-powered sensor nodes in smart agricultural applications. However, the buck-boost unit in HEHD minimized the matching impedance and led to a time delay.

Lili Xia [22] introduced a hybrid energy harvester which was a combination of solar and wind with an oscillation-induced function. The hybrid harvester utilized a vibrational device to create electrical energy. When the harvester is placed in good lightening conditions, it helps maintain a stable state and optimize the power efficiency. Similarly, during poor lighting conditions, the harvester utilized a specified control unit to clean the Photovoltaic (PV) panels. However, the introduced hybrid harvester requires a high cost to maintain the PV panels at regular time intervals.

Lei Hu et al. [23] developed an electromechanical quality of a hybrid broadband wind energy harvester to monitor smart agriculture in the loess plateau. A wide-band Energy Conversion Device (ECD) that combines a Triboelectric Nanogenerator (TENG) with an Electromagnetic Generator (EMG) was implemented to collect wind energy across various wind speeds effectively. To optimize the wind energy usage by TENG and minimize energy dissipation, the team implemented an optimized Scotch yoke mechanism. An extension of the deflector into the fan significantly diminished the start-up wind speed and enhanced a device's capability to capture wind energy. In order to ensure stability and extend an ECD service life, an omnidirectional conductive foam was utilized as both electrode and friction substance. This method should focus on stable wind patterns because wind patterns may vary throughout different seasons and locations, which could impact the harvester's overall energy generation and reliability.

Pengfei Chen et al. [24], a Fur-Brush Triboelectric Nanogenerator (FB-TENG) was implemented to harvest water and wind energy in smart agriculture. This method utilized naturally available animal furs in the FB-TENG due to their exceptional characteristics, including exceedingly low wear, great performance, and resistance to humidity. Compared to conventional TENGs, FB-TENG demonstrated remarkable durability and robustness even at low driving torque, durability to minimal friction and wear caused by gentle fur material. FB-TENG maintained a consistent output efficiency regardless of fluctuations in environmental humidity due to the density of its furrows. It exhibited stability even when exposed to a wide range of humidity levels, ranging from 40% to 90%. Through the utilization of a counter-rotating structure, the relative rotation of the fur disk and electrode disk in an FB-TENG was able to effectively amplify the output current and significantly enhance the overall output power. However, the energy generation potential of the FB-TENG may be limited in dry regions or during drought periods. To ensure optimal

performance, a sufficient and consistent water source was required for water-based energy harvesting in the utilization of TENG methods.

Xinqing Xiao et al. [25] developed a temperature monitoring method to store food based on solar energy harvesting and wireless charging. A method integrated wireless temperature sensing, solar energy harvesting and wireless charging capacities. The performance of wireless charging among sensor nodes and transmitters was analyzed, along with the energy utilization of the wireless sensor node. The performance of the wireless temperature monitor was evaluated; the formation and optimization of wireless charging and solar energy harvesting-based temperature monitoring methods were also assessed. An implemented system efficiently enables real-time wireless temperature monitoring by utilizing wireless charging and solar energy harvest, ensuring food's condition and safety during storage. However, in this method, food storage containers or devices should be monitored in close proximity to the charging station; for effective charging and temperature monitoring, the system had a defined charging range.

4. Harvesting System using IoT Technology

The hybrid energy harvesting system is intended to be based on IoT technology which can collect more data from vast areas. This paper utilized a hybrid energy harvesting system which is a combination of solar, wind and grid harvesters. Figure 1 depicts the schematic process involved in energy harvesting.

At the initial stage, the grid stations are used to produce power of 230v, and the transfer of energy from one source to other devices takes place using an isolation transformer. Moreover, the conversion of AC to DC takes place, and additional noises are removed using rectifiers and filters. The solar energy is harvested using 12 volts of solar photovoltaic cells, which are used to generate the energy per unit area obtained from the sun in the form of electromagnetic radiations, where these photovoltaic cells directly convert the energy into DC electric energy. Wind energy uses the wind to produce kinetic energy through wind turbines to turn electric generators for electric power. The wind machine torque changes its rates of the angular moment based on the wind speed and pitch angle. The useful load of energy generated by wind on the rotor is transmitted to the asynchronous machine and generates AC electricity. The universal Bridge converts this AC electricity to DC electricity and transmits it to the booster. The energy from solar, wind and grid harvesters is stored in 200 volts, 6.5Ah Ni-MH battery and then transmitted to the various nodes. At the initial stage, the sensors are plotted in the agricultural field on every corner of the field; these sensors emit UV light attracting pests and insects towards it. The sensors which emit UV light are charged using an energy harvesting model, which is a combination of grid, solar and wind energy.
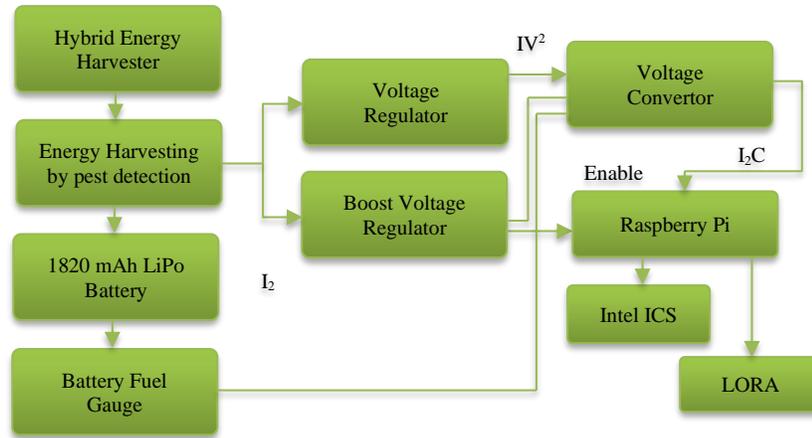
Fig. 1 Block diagram of the process involved in hybrid energy harvesting

The grid station generates a constant 230v at 50 frequencies. The solar photovoltaic cells generate 12 volts. The energy harvested from the three sources is stored in an 1820 mAh LiPo battery. According to the count of the total number of pests in the sensor area, the voltage passes through a voltage regulator and the voltage is converted from AC to DC using a voltage converter. The presence of pests is captured using an image sensor (Sony IMX219) which captures the pests' image and classifies the insects using bounding boxes. The red box indicates the presence of codling moths (harmful insects), and the blue box indicates the presence of general insects (non-harmful insects). Thus the model effectively classifies the presence of insects and helps in precise agriculture.

4.1. Implementation of Hardware Components
Each trap is designed based on a customized platform which consists of a low-power image sensor to gather images, Raspberry Pi from a single board computer, and a hybrid energy harvester for collecting and storing the energy.

4.1.1. Sensing
The image sensor called Sony IMX219 is utilized, which is a low-power back-illuminated image sensor. This has been made up of 1.12μm pixel technology, which is highly sensitive and requires a minimal number of components. The image sensor combines the circuit at black level calibration and helps to lower the computation. Moreover, it instructs the sensors to minimize the consumption of power.

4.1.2. Processing
The pre-processing takes place in two stages; at the initial stage, the Raspberry minicomputer is used to regulate the sensor's acquisition and helps process the images. The Pi3 version is utilized in this process due to its computing ability, energy demand, and lower cost. The next stage contains a neural accelerator known as Intel Neural Compute stick which reduces the inference time.

4.1.3. Transmission
The Long-Range modulation is equipped with the smart trap, and the connection is offered with the help of the RFM95W transceiver. The 2dB gain is obtained during the connection of LoRa IC with an antenna.

4.1.4. Power Supply
The power supply for the whole integrated system is provided by a LiPo battery. The combination of solar, wind, and grids are directly connected with the energy harvester, which charges the Li-Po battery with 1820 mAh. The power supply contains two types of converters connected to it. The first one is regulated by MCP1812, which generates 3.3V to a microcontroller. The second one is the Boost converter which offers a stable power of 5v to the Raspberry Pi. A battery fuel gauge is utilized for the process of monitoring the condition of the battery. Moreover, the battery fuel gauge is responsible for handling the harvesting process by providing better battery life without the presence of the farmer's supervision.

4.2. Pipeline for Region Detection
The pipeline is defined as the collection of components utilized during the processing of the flow of data in its manner. After capturing the image and segregating the harmful pests, the smart trap performs the multi-stage process. The automated pipeline for the detection of pests is represented in Figure 2.

The pipeline process utilizes sliding windows and trained image classifiers. The classifiers are utilized in the concept of sliding windows at various Regions of Interest (ROI). A part of the captured image of an insect is known as ROI. The regular arrangement is performed densely over the ROI, and a large amount of overlapping occurs above the image. The overlapping that occurred over the images can be rectified using a Gaussian filter which effectively smoothens the image by removing unnecessary pixels.
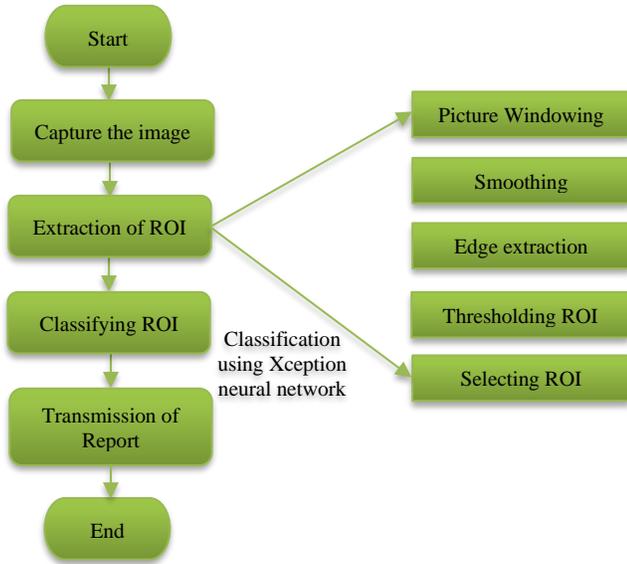
```
        ┌─────────┐                    ┌──────────────────────┐
        │  Start  │                    │  Picture Windowing   │
        └─────────┘                    └──────────────────────┘

   ┌──────────────────┐                ┌──────────────────────┐
   │ Capture the image│                │     Smoothing        │
   └──────────────────┘                └──────────────────────┘

   ┌──────────────────┐                ┌──────────────────────┐
   │ Extraction of ROI│                │   Edge extraction    │
   └──────────────────┘                └──────────────────────┘

   ┌──────────────────┐                ┌──────────────────────┐
   │  Classifying ROI │  Classification│   Thresholding ROI   │
   └──────────────────┘  using Xception└──────────────────────┘
                         neural network
   ┌──────────────────┐                ┌──────────────────────┐
   │ Transmission of  │                │     Selecting ROI    │
   │     Report       │                └──────────────────────┘
   └──────────────────┘

        ┌─────────┐
        │   End   │
        └─────────┘
```

Fig. 2 Flow diagram of detection of pest

The filtered image undergoes the edge extraction process, which is performed using a canny filter that selects the ROI with maximum probability. After the detection stage, ROI is analysed by an algorithm, namely Principal Component Analysis (PCA) algorithm, to detect the pests.

4.3. Edge Accelerator

The edge accelerator is generated based on a System on a Chip (SoC) aided specifically for the DL models. Various corporations have produced different hardware to improve the performance of algorithms related to deep learning. This research analyzed three kinds of platforms such as Intel NCS2, Nvidia Jetson Nano, and Google Coral USB TPU.

4.3.1. Consumption of Energy

Energy is considered one of the significant sources of battery-powered edge accelerators. The Jetson Nano is the most power-hungry platform in terms of energy consumption, requiring up to 10W when utilizing the GPU during inference. On the other hand, Google TPU consumes about 5W of electricity. The power usage is comparable for Intel NCS, which consumed 2W for the accelerator and 3W for Raspberry Pi.

Maximum Power Point Tracking (MPPT)

This research utilized hardware-based MPPT to attain energy neutrality. MPPT comprises a switching mode power converter and Maximum Power Point (MPP) controller to regulate the impedance and provides maximum power to the energy buffer and the load. MPP attains this variation by evaluating the voltage and current. For the DC sources like solar energy and wind energy, the MPP is utilized as a combination of voltage and current that enhance energy efficiency and helps to achieve neutrality. The MPPT determines voltage and current, which is used to evaluate the input intensity of the energy harvesting model. For example, the MPP of solar energy is determined by the intensity of the solar light and the temperature.

Similarly, for wind energy, the rotational speed sensor is utilized. Energy neutrality is attained only when the harvested power is higher than the minimum power consumption of the system. Additionally, in most hardware-based MPPTs, maintaining a low overhead is an easy task. The overall process involved in Maximum Power Tracking Algorithm is presented in Figure 3.

4.3.2. Performance

Model inference time for execution is an important statistic for sensory systems. The Nvidia Jetson has the most processing capability of the three systems studied, tailed by Google TPU. Less power consumption takes place using Intel NCS 2, and it is fully adequate for the suggested solution, which does not need hard real-time ML job execution. Because this research is focused on energy saving, the Intel NCS2 was chosen to operate as a neural accelerator for the suggested application.

4.3.3. Availability

The systems and configurations evaluated were also limited by hardware availability. The design space investigation includes analyzing the three neural accelerators. Among the three, Intel NCS2 provides optimal trade-offs in its performance, consumption of energy, and compatibility.

4.3.4. Principle Component Analysis

The Principal Component Analysis (PCA) is a significant process in the orthogonal linear transformation of the ROI. PCA highlights the dissimilarity and removes the unnecessary part of the image. Moreover, it detects the mean value of the data and evaluates the principal components. PCA is best suited for performing extracting the essential features from the ROI image. PCA is utilized for different operations on the image matrix to convert it to a low-dimensional Eigen subspace. The Eigenvectors with maximum value are deliberated as principal components. The steps involved in the PCA algorithm are presented below:

1. Collect the input data from the ROI
2. Evaluate the mean value
3. Subtract the mean from every individual data
4. Evaluate the Eigenvector and the Eigenvalues
5. Detect the maximum Eigenvalue
6. Compute the weight

The set of Eigenvectors is known as Eigen's face, which is utilized in the process of detecting the image of pests. This Eigen's face extracts the significant features from the image obtained from ROI. In this research, K images are obtained which is based on the number of trained images from ROI and the loaded images are denoted as L, which is presented in Equation (1) as follows:
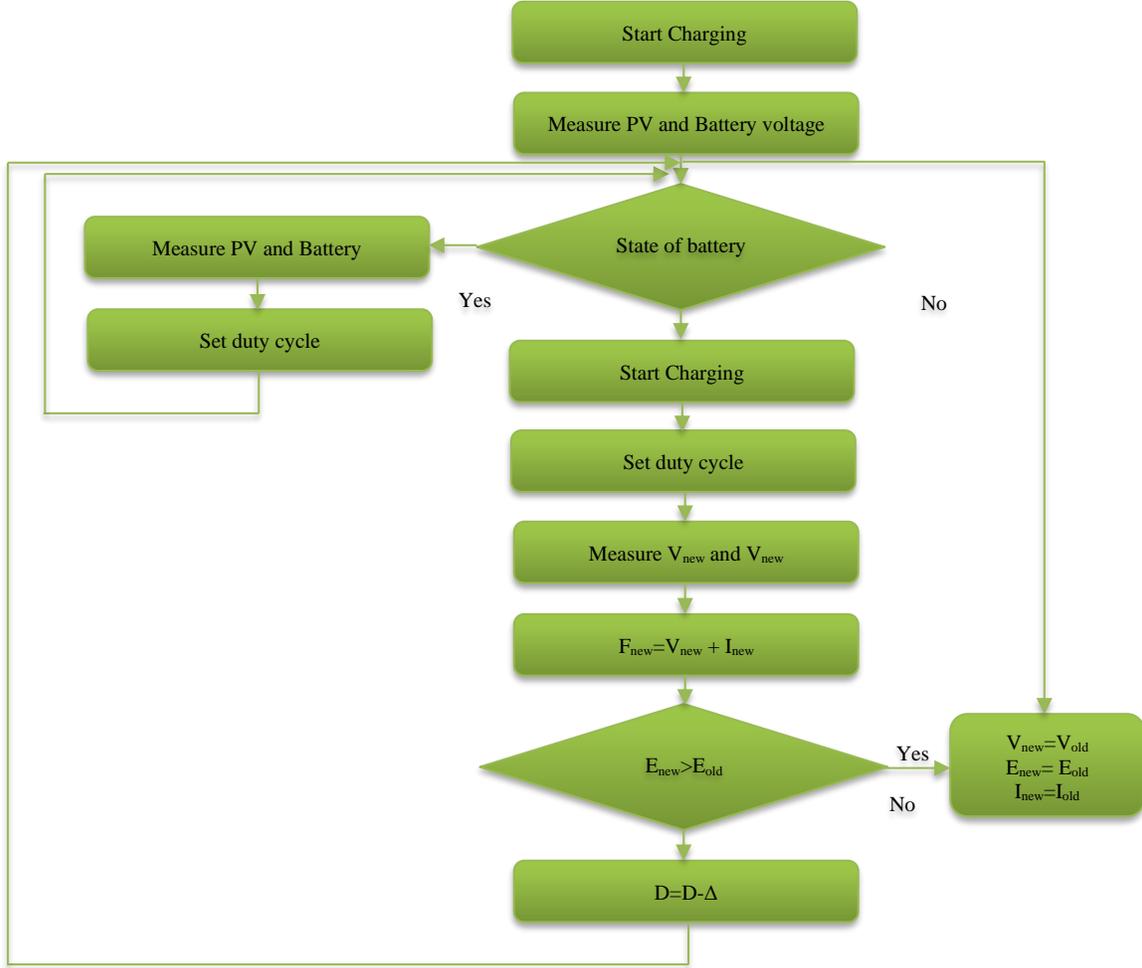
Fig. 3 Flowchart of MPPT

$$L = \{\Gamma_1, \Gamma_2, \dots, \Gamma_K\} \tag{1}$$

Where L is the loaded image and the trained K image is denoted as Γ_K.

After the process of loading, the next step is to evaluate the mean value, which is denoted as $\bar{X}$ and it is evaluated using Equation (2) as follows:

$$\bar{X} = \frac{1}{K} \sum_{n=1}^{n=K} \Gamma_K \tag{2}$$

Then subtract the mean face from each image which is known as normalization. The outcomes obtained from the normalization process get stored in a new matrix. This step is mathematically represented in Equations (3) and (4) as follows:

$$\phi_n = \Gamma_K - \bar{X} \tag{3}$$

$$D = \phi_1, \phi_2, \dots, \phi_n \tag{4}$$

Where ϕ_n is the variable of normalization, $\bar{X}$ is the average face, and D is the matrix.

After the stage of normalization, a normalized face vector is obtained. Then, the matrix co-variance is evaluated for the normalized vector, which is presented in Equation (5) as follows:

$$C = DD^T s \tag{5}$$

Where C is the co-variance matrix of matrix s.
The effective calculation of the Eigenvalue provides Eigenvector with a reduced space vector. For the process of image detection, the weight of the image is evaluated based on Equation (6) as follows:

$$\omega_K = \mu^T (\Gamma_K - \bar{X}) \tag{6}$$

Where ω is the weight and the Eigenvector is denoted as μ.

4.4. Xception Neural Network

The weighted image obtained from the PCA algorithm proceeded with the classification process. In this research, the Xception neural network is used for classifying pests. The Xception Neural Network is an enhanced form of the

InceptionV3 model, which is a structure of the deep neural network and is depth-wise separable. There are about 36 convolutional layers present, which are categorized into 14 blocks. The depth-wise separable layers present in the Xception model undergo convolution at every individual layer for each channel. The Xception takes place as an ordinary convolution process and performs 3×3 operation after the completion of 1×1 convolution. The architecture of the Xception model is presented in Figure 4.

In Figure 4, initially, the data travels through the entrance gateway and to the middle gateway, where the process is repeated eight times and then goes over the exit gateway. Normalization must be done for all the convolutional layers and the separable layers. For a standard convolutional layer, the input size of the feature map is considered as $D_F \times D_F \times M$ and the size of the feature map at the output side is assumed as $D_F \times D_F \times N$. Where the width and height of the spatial map are represented as D_F. The number of input channel and the number of output channel is represented as M and N, respectively. The formula utilized for the computation of the output feature is represented in Equation (7) as follows:

$$G_{k,l,n} = \sum_{i,j,m} K_{i,j,m,n} . F_{k+i-1,l+j-1,m} \qquad (7)$$

Where the depth convolutional kernel is denoted as K, and the output of the feature map is represented as G. The cost for computation of standard convolution is represented as follows,

$$D_k . D_k . M . N . D_F . D_F$$

Where the spatial dimension of the kernel is denoted as D_k.

The depth of the convolution for each input channel is represented in Equation (8) as follows,

$$\hat{G}_{k,l,n} = \sum_{i,j,m} \hat{K}_{i,j,m,n} . F_{k+i-1,l+j-1,m} \qquad (8)$$

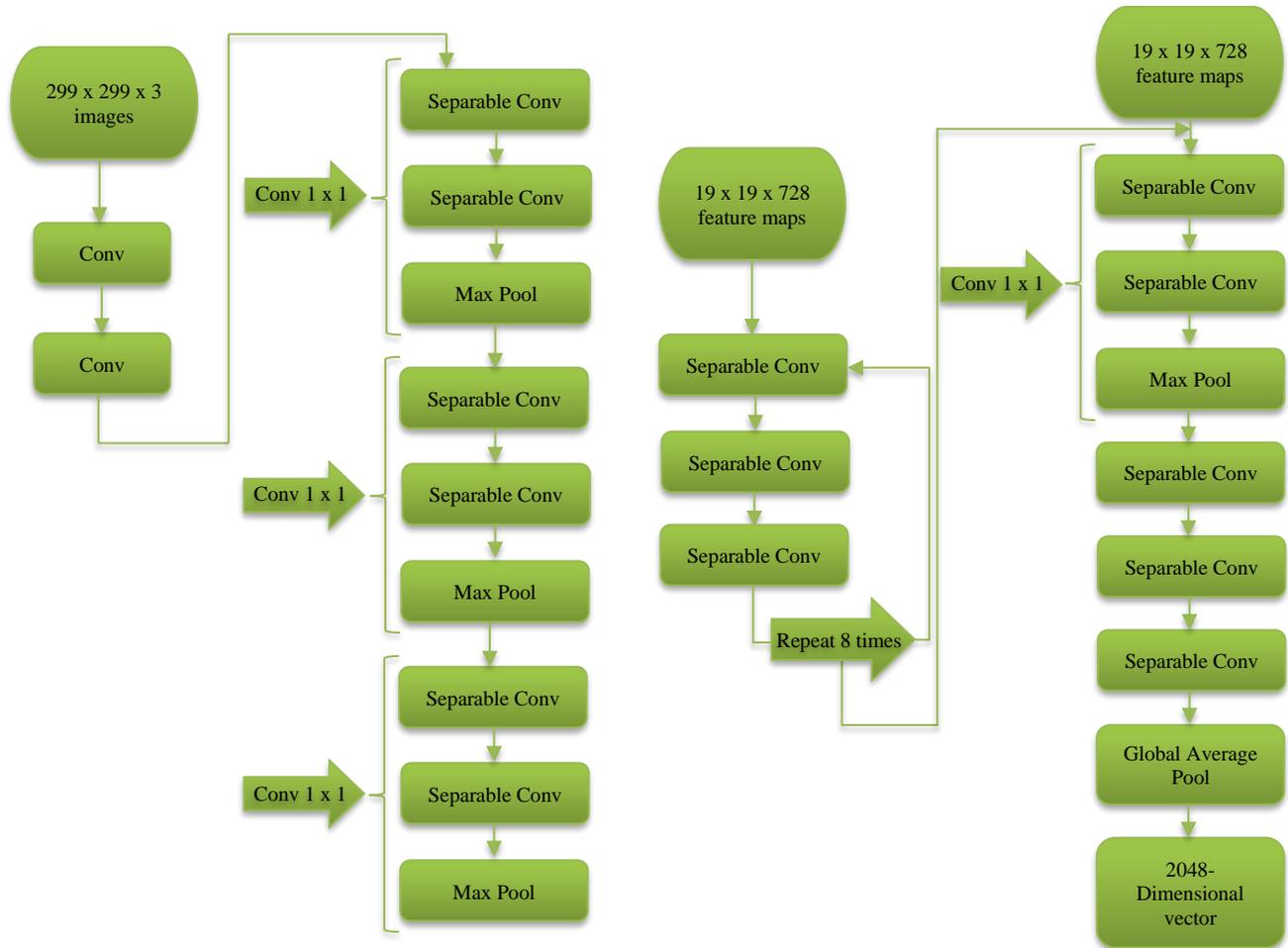Where the filtered feature map is denoted as $\hat{G}$ and the filtered convolutional kernel is denoted as $\hat{K}$.

Fig. 4 Architectural diagram of the Xception model

The training of the Xception model is performed using the categorical cross-entropy function, which is evaluated in Equation (9) as follows,

$$loss = -\sum_{i=1}^{n} \hat{y}_{i1} log y_{i1} + \hat{y}_{i2} log y_{i2} + \cdots + \hat{y}_{im} log y_{im}$$

(9)

Where the number of samples is denoted as n, and the number of classifications is denoted as m.

4.4.1. Training the Xception Model

The dataset is trained in an automated way by image processing technique. The relevant features are extracted from the raw images and created tiles with the insects and codling moths. The dataset consists of around 1100 images, and 30% of them are utilized for the process of validation. The images present in the dataset can be enhanced by integrating the augmentation techniques prior to training. The model is trained for about 100 epochs where the size of images is around 52×52. During the training process, the Xception model performed better training without overfitting.

4.4.2. Augmentation of Data and Optimization of Network

Building bigger and larger neural networks have contributed significantly to deep learning's effectiveness. This helps the models to perform better on certain tasks, but it also increases their cost of usage. Larger models require more storage space, making them more difficult to allocate. Higher versions take longer to run and may need more expensive hardware. The optimization step seeks to minimize the model's size while minimalizing accuracy or performance. This enables faster examination while avoiding loss in accuracy. The suggested execution optimizes the model in the period of training and before training the model.

4.4.3. Augmenting the Data

In this research, data augmentation is performed to enhance the count of images during training. The images of the trap from the drone view show that the images will not modify the labels' classes during the rotation or flipping period.

4.4.4. Pruning

The neural network's pruning is one technique where insignificant neurons are removed from the trained model. The pruning can be performed in various ways, such as pruning the weights by citing the separate constraints to zero, and entire nodes present in the network can be removed from the network to maintain the accuracy of the large network at their initial stage. Pruning helps to enhance the accuracy of the network. Pruning the network takes place during the training period and helps achieve the required accuracy during the validation stage.

4.4.5. Optimization

The optimization is performed after the stage of training where the complexities present in the network are lowered and enhance the evaluation speed. The faster inference of the deep learning model is performed by merging the nodes, constant horizontal fusion, and normalization of batches. Moreover, the unused layers present in the network are dropped out in the stage of training.

5. Results and Analysis

In this research, the occurrence of codling moths is monitored twice a day. The flow of the process in the smart trap is represented in the following steps.

- Initially, the power is switched ON (start)
- The image of trapped insects is captured (P1)
- The captured images are pre-processed (P2)
- Next, the classification algorithm is executed (P3)
- The results were sent using the radio (P4)
- After this process, the power is switched OFF (End)

These steps are utilized to describe the performance of smart traps in energy consumption and accuracy. The image of the evaluated photo is shown in Figure 5, where the codling moths are bounded by red boxes and the other insects are bounded by blue boxes.

The results are analysed based on the performance of the Xception model with other models, and the comparison is performed with the existing energy harvesting models discussed in the related works.
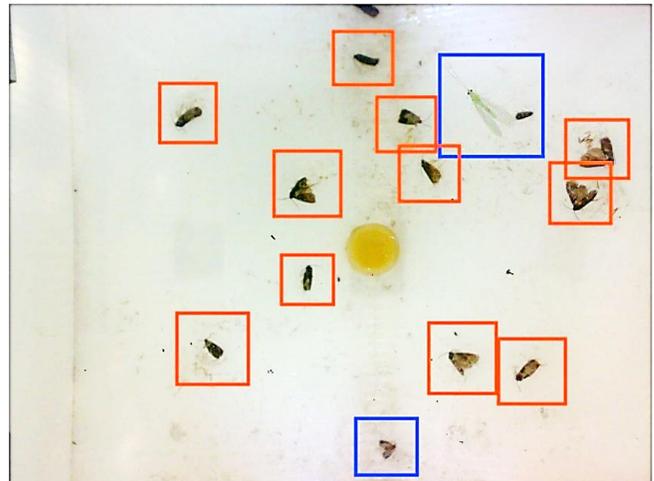

Fig. 5 Image of annotated photo

5.1. Experimental Setup

The proposed model is built with hardware components such as an image sensor called Sony IMX219 for sensing, RFM95W transceiver, Raspberry Pi3 and Li-Po battery with 1820 mAh to store the power. Moreover, the implementation of the proposed research is done using MATLAB. The experimental setup of Raspberry Pi3 with Sony IMX219 image sensor is shown in Figure 6 as follows:

Table 1. Energy consumption of the networks

Process	Raspberry Pi MobileNetv2	Raspberry Pi LeNet	Raspberry Pi VGG16	Raspberry Pi Xception	Raspberry Pi MobileNetv2 with Intel NCS	Raspberry Pi LeNet with Intel NCS	Raspberry Pi VGG16 with Intel NCS	Raspberry Pi Xception with Intel NCS
Start(J)	40.7	40.7	40.7	40.7	40.7	40.7	40.7	40.7
P1(J)	2.171	1.674	1.733	1.423	2.427	2.177	2.359	2.121
P2(J)	6.125	5.453	6.491	4.632	6.923	7.054	7.041	7.012
P3(J)	119.1	57.4	114.3	53.5	70.34	59.04	73.53	55.34
P4(J)	2.147	2.147	2.147	2.147	2.273	2.273	2.273	2.273
End(J)	15.85	15.85	15.85	15.85	21.97	21.97	21.97	21.97
Total(J)	186.1	123.2	181.2	118.2	144.6	133.2	147.9	129.4
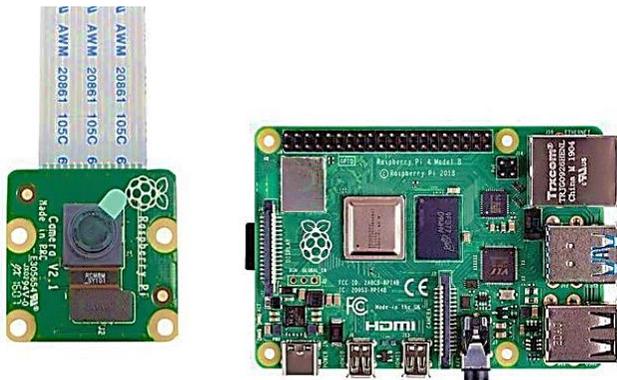
Fig. 6 Experimental setup of Raspberry Pi3 with image sensor

5.2. Performance Analysis

This subsection helps to evaluate the performance of the Xception model with the existing models such as LeNet-5, VGG-16, and Mobile NetV2. The performance is evaluated based on the energy consumption rate, accuracy, recall, precision and F-score. The energy consumption of the aforementioned networks is compared with the Xception net in Table 1 as follows.

The Raspberry Pi3 is utilized in evaluating the networks. Moreover, the performance is individually analysed with the presence and absence of Intel NCS. The energy consumption of the Li-Po battery with 1820 mAh with RPi3 is evaluated and tabulated as follows. The results from Table 1 show that the Xception model consumed minimum energy and helped enhance the Li-Po battery's lifespan with 1820 mAh. The Zception model consumed minimal energy of 118.2J and 129.4 J with Intel NCS. The Xception model undergoes convolution for depth-wise layers of the network. Moreover, it can perform large image classification for large datasets.

The performance of the classifiers is valued in terms of training accuracy and validation accuracy, as shown in Table 2. Xception model proceeded for training accuracy and validation accuracy. The graphical representation for LeNet, VGG 16, MobileNetV2 and Xception net is represented in Figure 7.

**Table 2. Analysis of accuracy for various classifiers**

Epochs	Training Accuracy	Validation Accuracy
LeNet-5		
1	0.81	0.8
2	0.84	0.81
3	0.8	0.82
4	0.89	0.83
5	0.83	0.83
6	0.9	0.91
7	0.92	0.92
8	0.94	0.93
VGG 16		
1	0.91	0.97
2	0.95	0.94
3	0.97	0.98
4	0.96	0.96
5	0.97	0.98
6	0.98	0.99
MobileNetV2		
1	0.81	0.8
2	0.84	0.81
3	0.8	0.83
4	0.89	0.84
5	0.83	0.84
6	0.93	0.92
7	0.94	0.95
8	0.95	0.96
Xception Net		
1	0.7	0.69
2	0.7	0.69
3	0.81	0.83
4	0.82	0.84
5	0.93	0.96
6	0.97	0.99

Table 3. Performance evaluation

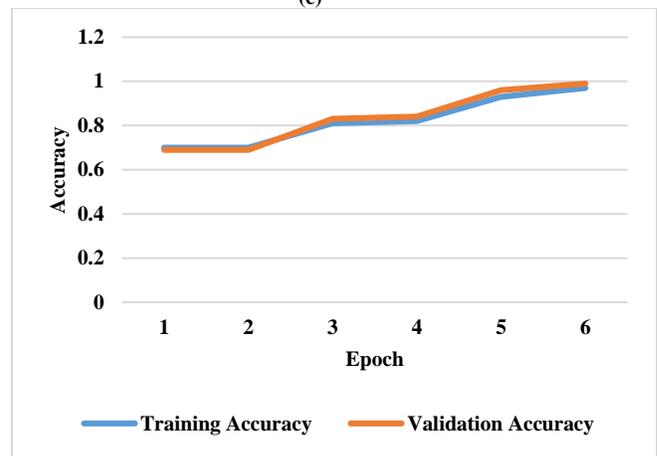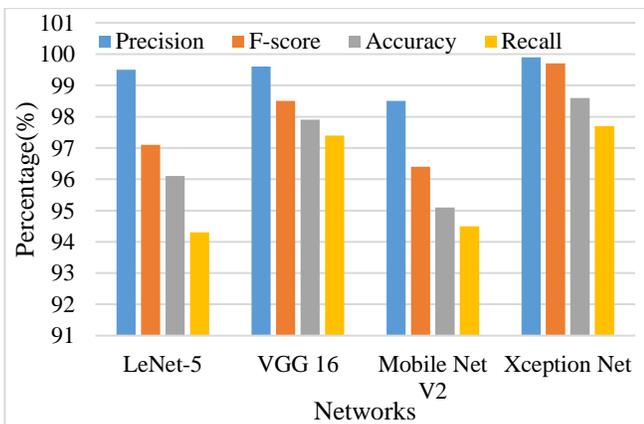| Networks	Precision (%)	F-score (%)	Accuracy (%)	Recall (%)
LeNet-5	99.5	97.1	96.1	94.3
VGG 16	99.6	98.5	97.9	97.4
Mobile Net V2	98.5	96.4	95.1	94.5
Xception Net	99.9	99.7	98.6	97.7
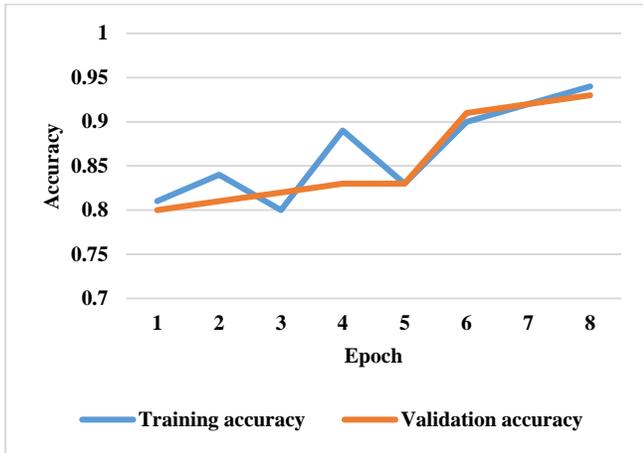
(a)
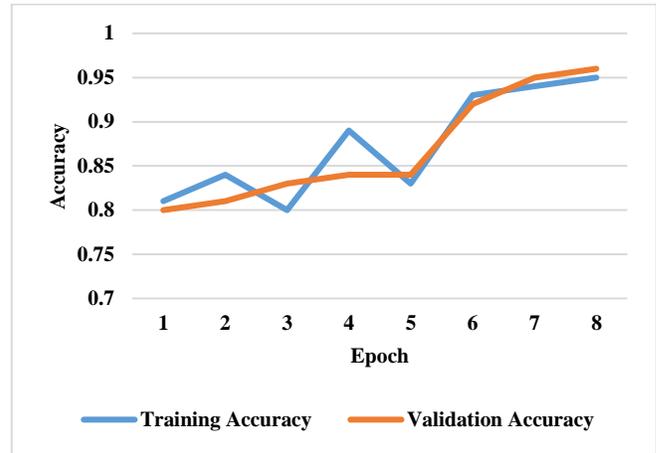
(c)
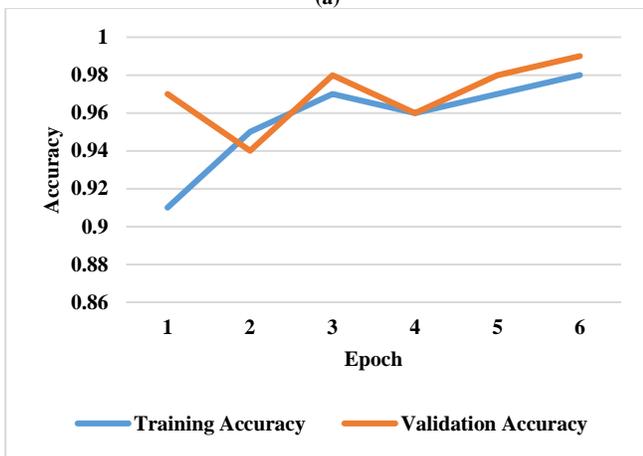
(b)

(d)

Fig. 7 Graph for comparison of training and validation accuracy of (a) LeNet-5, (b) VGG 16, (c) MobileNetV2, and (d) Xception Net

Fig. 8 Graphical representation of networks in terms of performance metrics

Similarly, the performance of the Xception model is evaluated for accuracy, recall, precision and F-score. Table 3 symbolizes the performance of the Xception model when compared with LeNet-5, VGG 16 and Mobile Net V2. The results of Table 3 are graphically represented in Fig 8.

The overall results from Table 3 show that the Xception model achieved better performance in all metrics compared to the remaining networks. It has attained a classification accuracy of 98.6% while remaining networks such as LeNet-5 (96.1%), VGG 16 (97.9%) and Mobile Net V2 (95.1%). The depth-wise convolution layer present in the Xception model helps to attain better performance when compared with other network models.

Table 4. Comparative table

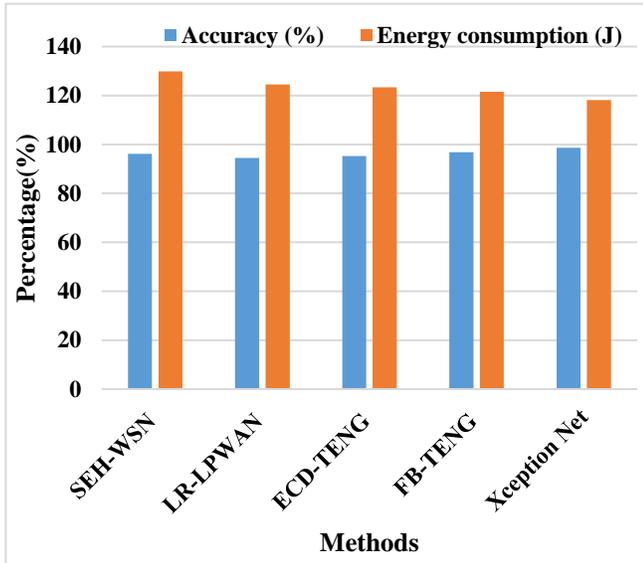| Methods	Accuracy (%)	Energy consumption (J)
SEH-WSN [16]	96.2	129.8
LR-LPWAN [18]	94.5	124.5
ECD-TENG [23]	95.3	123.3
FB-TENG [24]	96.8	121.6
Xception Net	98.6	118.2

Fig. 9 Comparison analysis of the proposed method with existing methods

5.3. Comparative Analysis

In this subsection, the effectiveness and efficiency of the Xception model are compared with the existing models utilized in harvesting energy. The comparison is performed for parameters such as accuracy and energy consumption rate. The efficiency of the Xception model is compared with Smart Energy Harvesting using Wireless Sensor Networks (SEH-WSN) [16], Long Range – Low Power Wide Area Networks (LR-LPWAN) [20], Energy Conversion Device that combines a Triboelectric Nanogenerator (ECD-TENG) [23], and Fur-Brush Triboelectric Nanogenerator (FB-TENG) [24]. Table 4 represents the accuracy and energy consumption of the Xception model when compared with SEH-WSN, LR-LPWAN, ECD-TENG, and FB-TENG.

The methods such as SHE-WSN [16], LR-LPWAN [20], ECD-TENG [23], and FB-TENG [24] are evaluated for the same experimental setup, which is mentioned in section 4.1 to compute the effectiveness of the proposed method. The results of Table 4 are represented graphically, as shown in

Fig 10. The results from Table 4 show that the Xception Net achieved a better accuracy value of 98.6% while SEH-WSN, LR-LPWAN, ECD-TENG, and FB-TENG attained an accuracy of 96.2%, 94.5%, 95.3%, and 96.8%. Moreover, the Xception model consumed less energy of 118.2 J which is comparatively lower than SEH-WSN (129.8 J), LR-LPWAN (124.5 J), ECD-TENG (123.3), and FB-TENG (121.6). The depth-wise convolution layer present in the Xception model helps in accurately classifying the codling moth and general insects.

6. Discussion

The SEH-WSN [16] has achieved 129.8J of energy; however, this method has high energy consumption compared to the proposed method and also has power management issues. These are resolved with the proposed method's MPPT mechanism, which neutralizes the energy for efficient harvesting. The LR-LPWAN [18] has achieved 124.5J of energy, but this method has an interface issue along with the improper selection of noes. The ECD-TENG [23] and FB-TENG [24] have achieved 95.3% and 96.8% accuracy, 123.3J and 121.6 J energy consumption. These two methods have the drawback of low system stability in generating energy. These drawbacks are addressed in the proposed research with the use of PCA and Raspberry Pi3.

7. Conclusion

Hybrid energy harvesting is envisioned to be a critical component to achieving energy neutrality for the Internet of Things-based Smart Agricultural Systems. This research introduced a hybrid energy harvesting model based on IoT technology. The hybrid energy harvesting model combines solar, wind, and grid harvesters. The Xception network is utilized to evaluate the images of harmful and non-harmful insects. The detection of pests is performed using the high-level hardware configuration using Raspberry Pi3. This research helps the agricultural sector and economically improve farmers' status by providing better yields. The results obtained from the experiments show that the proposed method has achieved better accuracy of 98.6%, which is comparatively higher than the existing SHE-WSN, LR-LPWAN, ECD-TENG, and FB-TENG with 96.2%, 94.5%, 95.3%, and 96.8% respectively. In the future, research can be proceeded to reduce the energy in the process of pest detection using Raspberry Pi 4.

Author Contributions

All authors have contributed equally to conceptualization, methodology, validation, resources, writing—original draft preparation, review and editing for this research work.

References

[1] Mobasshir Mahbub, "A Smart Farming Concept Based on Smart Embedded Electronics, Internet of Things and Wireless Sensor Network," *Internet Things*, vol. 9, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[2]   Elsayed Said Mohamed et al., "Smart Farming for Improving Agricultural Management," *The Egyptian Journal of Remote Sensing and Space Science,* vol. 24, no. 3, pp. 971-981, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[3]   Abdul Salam, "Internet of Things in Agricultural Innovation and Security," *Internet of Things for Sustainable Community Development*, pp. 71-112, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[4]   Lingyi Lan et al., "Breathable Nanogenerators for an On-Plant Self-Powered Sustainable Agriculture System," *ACS Nano*, vol. 15, no. 3, pp. 5307-5315, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5]   Omer Gulec, Elif Haytaoglu, and Sezai Tokat, "A Novel Distributed CDS Algorithm for Extending Lifetime of WSNs with Solar Energy Harvester Nodes for Smart Agriculture Applications," *IEEE Access*, vol. 8, pp. 58859-58873, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[6]   Achilles D. Boursianis et al., "Smart Irrigation System for Precision Agriculture-The AREThOU5A IoT Platform," *IEEE Sensors Journal*, vol. 21, no. 16, pp. 17539-17547, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[7]   Xunjia Li et al., "A Self-Charging Device with Bionic Self-Cleaning Interface for Energy Harvesting," *Nano Energy*, vol. 73, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[8]   Sajib Roy, "An Electromagnetic Wind Energy Harvester Based on Rotational Magnet Pole-Pairs for Autonomous IoT Applications," *Energies*, vol. 15, no. 15, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[9]   Mahendra Swain et al., "LoRa-LBO: An Experimental Analysis of LoRa Link Budget Optimization in Custom Build IoT Test Bed for Agriculture 4.0," *Agronomy*, vol. 11, no. 5, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[10]   Demin Gao et al., "A Framework for Agricultural Pest and Disease Monitoring Based on Internet-of-Things and Unmanned Aerial Vehicles," *Sensors*, vol. 20, no. 5, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[11]   Olalekan Kunle Ajiboye et al., "Hybrid Renewable Energy System Optimization via Slime Mould Algorithm," *International Journal of Engineering Trends and Technology*, vol. 71, no. 6, pp. 83-95, 2023. [CrossRef] [Publisher Link]

[12]   Gaia Codeluppi et al., "LoRaFarM: A LoRaWAN-Based Smart Farming Modular IoT Architecture," *Sensors*, vol. 20, no. 7, p. 2028, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[13]   Ritesh Kumar Singh et al., "Leveraging LoRaWAN Technology for Precision Agriculture in Greenhouses," *Sensors*, vol. 20, no. 7, p. 1827, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14]   Eshwar Ravishankar et al., "Balancing Crop Production and Energy Harvesting in Organic Solar-Powered Greenhouses," *Cell Reports Physical Science*, vol. 2, no. 3, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[15]   Ying-Chih Lai et al., "Waterproof Fabric-Based Multifunctional Triboelectric Nanogenerator for Universally Harvesting Energy from Raindrops, Wind, and Human Motions and as Self-Powered Sensors," *Advanced Science*, vol. 6, no. 5, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[16]   Himanshu Sharma, Ahteshamul Haque, and Zainul Abdin Jaffery, "Maximization of Wireless Sensor Network Lifetime Using Solar Energy Harvesting for Smart Agriculture Monitoring," *Ad Hoc Networks*, vol. 94, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[17]   Murtaza Cicioğlu, and Ali Çalhan, "Smart Agriculture with Internet of Things in Cornfields," *Computers & Electrical Engineering*, vol. 90, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[18]   Weidang Lu et al., "Energy Efficiency Optimization in SWIPT Enabled WSNs for Smart Agriculture," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 4335-4344, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19]   Chu Donatus Iweh, Semassou Guy Clarence, and Ahouansou H. Roger, "The Optimization of Hybrid Renewables for Rural Electrification: Techniques and the Design Problem," *International Journal of Engineering Trends and Technology*, vol. 70, no. 9, pp. 222-239, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[20]   Tran Anh Khoa et al., "Smart Agriculture Using IoT Multi-Sensors: A Novel Watering Management System," *Journal of Sensor and Actuator Networks*, vol. 8, no. 3, p. 45, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[21]   Zhixin Wang et al., "Hybridized Energy Harvesting Device Based On High-Performance Triboelectric Nanogenerator for Smart Agriculture Applications," *Nano Energy*, vol. 102, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[22]   Lili Xia et al., "A Wind-Solar Hybrid Energy Harvesting Approach Based on Wind-Induced Vibration Structure Applied in Smart Agriculture," *Micromachines*, vol. 14, no. 1, p. 58, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[23]   Lei Hu et al., "Electromechanical Properties of a Hybrid Broadband Wind Energy Harvester for Smart Agriculture Monitoring in the Loess Plateau," *Electronics*, vol. 12, no. 1, p.34, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[24]   Pengfei Chen et al., "Super-Durable, Low-Wear, and High-Performance Fur-Brush Triboelectric Nanogenerator for Wind and Water Energy Harvesting for Smart Agriculture," *Advanced Energy Materials*, vol. 11, no. 9, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25]   Xinqing Xiao, Meng Wang, and Guoqing Cao, "Solar Energy Harvesting and Wireless Charging Based Temperature Monitoring System for Food Storage," *Sensors International*, vol. 4, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[26] Javier Rodríguez-Robles et al., "Autonomous Sensor Network for Rural Agriculture Environments, Low Cost, and Energy Self-Charge," *Sustainability*, vol. 12, no. 15, p. 5913, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[27] Himanshu Agrawal et al., "An Improved Energy Efficient System for IoT Enabled Precision Agriculture," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, no. 6, pp. 2337-2348, 2020. [CrossRef] [Google Scholar] [Publisher Link]