IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received May 25, 2022, accepted June 11, 2022, date of publication June 15, 2022, date of current version June 21, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3183277

Smart Crop Growth Monitoring Based on System Adaptivity and Edge AI

CHUN-HSIAN HUANG, (Member, IEEE), BO-WEI CHEN, YI-JIE LIN, AND JIA-XUAN ZHENG
Department of Computer Science and Information Engineering, National Taitung University, Taitung 950, Taiwan

Corresponding author: Chun-Hsian Huang (huangch@nttu.edu.tw)

This work was supported in part by the Research Project, Ministry of Science and Technology, Taiwan, under Grant MOST 107-2221-E-143-002-MY3 and Grant MOST 110-2634-F-194-006.

ABSTRACT This work proposes a smart crop growth monitoring system that contains an adaptive cryptography engine to ensure the security of sensor data and an edge artificial intelligence (AI) based estimator to classify the pest and disease severity (PDS) of target crops. Based on the smart system management mechanism, cryptographic functions can be adapted to varying and real-time requirements, while the actuators can be controlled to interact with the physical world to ensure the healthy growth of crops. Experiments show when all the four cryptographic hardware modules, including RTEA32, RTEA64, XTEA32 and XTEA64, are supported, using the adaptive cryptography engine, 72.4% of slice LUTs and 68.4% of slice registers in terms of the Xilinx Zynq-7000 XC7Z020 chip can be saved. Through the smart system management mechanism, a power consumption of 0.009 watts can be reduced. Furthermore, using the binarized neural network (BNN) hardware module of the PDS estimator, the recognition accuracy of target crops i.e. dragon fruits can achieve 76.57%. Compared to the microprocessor-based design and the GPU accelerated one, the same BNN architecture on the FPGA can accelerate the frames per second by a factor of 4,919.29 and a factor of 1.08, respectively.

INDEX TERMS Edge AI, binarized neural networks, adaptivity, crop growth monitoring, FPGA.

I. INTRODUCTION

Recently, the abnormal climate leads to the extreme weather, while the occurrence of natural disasters such as typhoon, rainstorm and severe drought gradually increases. This causes great casualties and serious damages to our properties and environment. For agriculture, the extreme weather also makes the growth of crops unstable, and the problem of food shortage thus becomes more and more serious. For all countries in the world, the food crisis has also become a very important issue.

Until now, most crops are still planted in the outdoor. This means the growth of crops will be affected by the weather easily. This also makes the yield and quality of farm crops unstable. Compared to the opening planting environments, recently, the greenhouse becomes a new alternative due to its controllable advantage. With the incoming of agriculture 4.0, new techniques such as cyber physical systems (CPS) [1] and Internet-of-Things (IoT) [2] further enhance the efficiency of the agricultural management. Furthermore, with the popularity of big data analytics [3], the trend of crop growth can be predicted and analyzed. For example, by applying sensors to the planting environment of crops, the collected data can be further analyzed to improve the productivity and quality of crops. Furthermore, the corresponding actuators such as sprinklers can be also controlled to interact with the physical world to ensure the healthy growth of crops.

For agriculture 4.0, there still exist some unsolved issues. To efficiently predict the crop yield, machine learning has been applied to the agricultural management [4]. Recently, with the popularity of deep learning, the convolutional neural network (CNN) [5] has been widely used to recognize the target crops and further evaluate their health statuses. However, the CNNs are very computing-intensive so that they are usually implemented on the powerful platforms such as servers. Furthermore, transferring the captured images to the server via the network also leads to high latencies on the Internet easily. In order to directly evaluate the health statuses of crops in the growth environment, integrating the deep learning techniques such as CNNs is thus required.

The associate editor coordinating the review of this manuscript and approving it for publication was Mario Donato Marino.

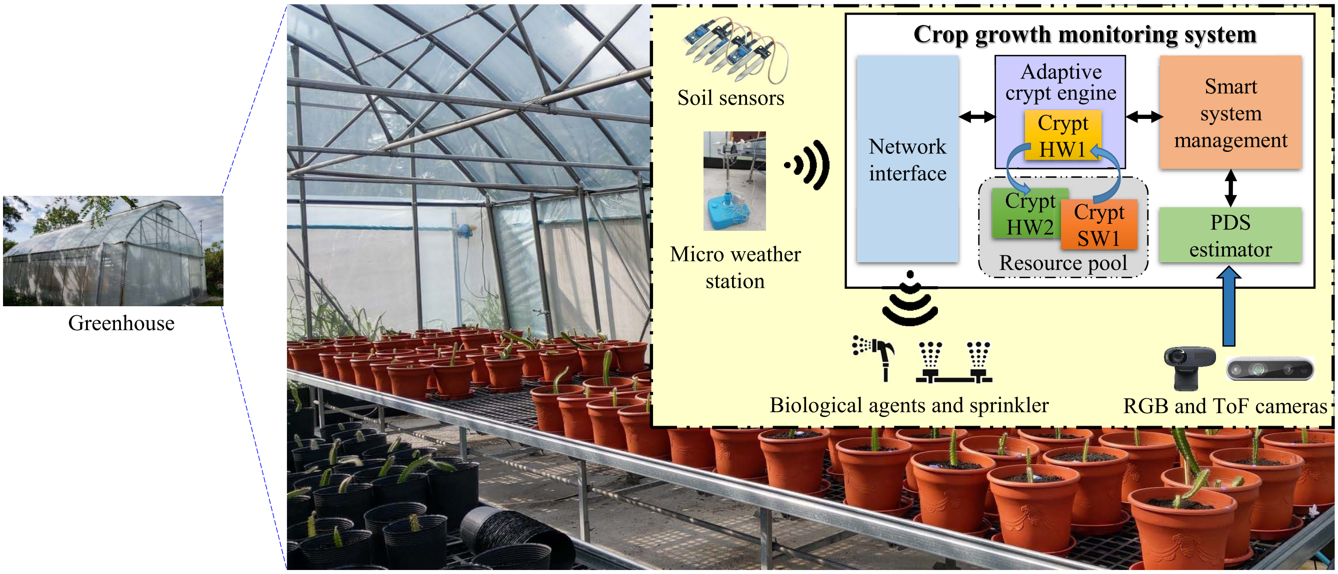

FIGURE 1. **Proposed crop growth monitoring system.**

Unfortunately, implementing the CNNs in the edge device always incurs a large amount of power consumption [6] and memory access [7]. For energy-constrained edge devices, this also becomes a very big challenge. Besides collecting the growth information about crops, the security of sensor data gradually attracts people's attention, especially when the collected data are related to commercial activities. As a result, the IoT security has recently become an important issue [8]–[10]. However, the sensor network in the growth environment of crops is usually unsupervised. Therefore, the security of sensor data cannot be ensured.

The motivation of this work is to evaluate the health statuses of crops in the growth environment directly, and thus the integration of the edge AI is necessary. To implement the edge AI design, system performance and power consumption are both the very important issues. Furthermore, the security of collected data is always neglected even though the collected data are related to commercial activities. However, until now, there is no design that can cover the smart, secure and scalable issues at the same time. To satisfy the above requirements, this work thus proposes a smart crop growth monitoring system based on system adaptivity and edge AI as shown in Figure 1.

The smart crop growth monitoring system is mainly applied to the greenhouse, where the soil sensors, the micro weather station, the RGB camera and the Time-of-Flight (ToF) one are used to monitor the target crops and their growth environment. Furthermore, this work is also to assist agricultural experts in developing biological agents to protect the crops from pests and diseases. As a result, the collected growth information of crops will be processed and analyzed for evaluating the effectiveness of developed biological agents. To ensure the security of sensor data, the cryptographic functions are integrated into the system. Furthermore, the cryptographic functions are implemented as reconfigurable hardware modules so that the smart crop growth monitoring system can not only support real-time data decryption but also adapt its cryptographic functions to varying requirements. To ensure the healthy growth of crops, a smart system manager is also presented to control actuators such sprinklers and biological agents to interact with the physical world. Figure 1 gives an ideal blueprint to apply the crop growth monitoring system to a real scene. However, it must solve the following issues.

1) How to integrate the CNN into the system to support the real-time estimation of the healthy statuses of crops?
2) How to make cryptographic hardware functions adaptable?
3) How to make the system smarter to ensure the healthy growth of crops?

For the above technical issues, in our smart crop growth monitoring system, new techniques are thus proposed. The novelty and contributions of this work are listed as follows.

- **Efficient estimation of PDS with edge AI and image fusion**: A resource-efficient binarized neural network (BNN) hardware module is presented to achieve the real-time recognition of the target crops, i.e. dragon fruits. Furthermore, based on the experience of agricultural experts, an image fusion method that leverages RGB and depth images is proposed to efficiently estimate the PDS of crops.
- **Adaptive protection of sensor data**: An adaptive cryptography engine is proposed to support not only real-time data decryption but also dynamically functional adaptivity. Furthermore, a layered and virtualizable architecture enables new user-designed cryptographic functions to be easily integrated into the system.
- **Agricultural cyber physical system**: Based on the estimation of PDS, a closed-loop control model is presented

to invoke the actuators to interact with the physical world to ensure the healthy growth of crops. Furthermore, based on the requirements of information protection in the connected sensor platforms, the crop growth monitoring system contains a system adaptation mechanism that can dynamically adapt its supported cryptographic functionalities.

This work is not a conceptual result. The proposed system is developed to provide the automation of the greenhouse and has been applied to a real scene. Thus, the agricultural experts can focus on the development of biological agents used for the protection of crop growth. This article is organized as follows. Section II describes the related work. Section III introduces our proposed crop growth monitoring system, including the edge AI based estimation of PDS, the adaptive cryptography engine and the smart system management mechanism. System implementation and evaluation are given in Section IV. Finally, Section V concludes this work.

II. RELATED WORK

For crop growth monitoring systems, besides the information collection of crop growth environments by using soil or micro weather sensors [11]–[16], the capture of crop images is another important task for a crop growth monitoring system. To estimate the PDS, calculating the size of a deformed or discoloured area relative to the whole crop is a typical method [17]. Furthermore, to recognize the target crops and further evaluate their health statuses, the machine learning has been widely applied to the system design. To identify plant diseases at the early stages, Nagasubramanian *et al.* [16] proposed an ensemble classification and pattern recognition for crop monitoring system, where ensemble nonlinear support vector machine was used to detect leaf and crop diseases. Zhang *et al.* [18] presented a method for monitoring the growth of greenhouse lettuce, in which a CNN model was used to learn the relationship between images and the corresponding growth-related traits, i.e., leaf fresh weight, leaf dry weight, and leaf area. Thangadeepiga and Raja [19] presented a CNN-based approach for remote sensing-based crop identification from Worldview-2 satellite data, and experiments showed the presented model yielded 78% accuracy for satellite image dataset and it provided 83% accuracy for field data collected. Furthermore, CNNs were also applied to the classification of paddy crops [20], the detection of fruits [21] and the localization of the picking points for a ridge-planting strawberry harvesting robot [22].

To apply the CNN to an edge device, by reducing the network sizes, quantized neural networks (QNNs) [23] were thus presented. The storage space and the bit-width of processing elements could be reduced through lower precision parameters, while system performance and throughput could be increased through the refined computations. This makes low-power deep learning applications possible [24]. Furthermore, for the hardware computing architecture to perform the CNN computation, the FPGA has recently become a new alternative because of the parallel architecture and energy efficiency [25]. Compared to CPUs, GPUs and ASICs, the FPGA-based designs [25]–[27] have been demonstrated that they were more advantageous to implement resource-efficient QNNs such as BNNs [24], [28], [29]. For agricultural applications, a low-power and high-speed deep FPGA inference engine performing the BNN [30] was a typical design at the edge for weed classification. The experiments also showed performing inference on weed images was 2.86 times faster than the best performing baseline full-precision GPU implementation. Besides the low-power and high-speed advantages, the reconfigurable ability of the FPGA enables new CNN architectures having different CNN parameters or topologies to be easily evaluated.

Besides collecting sensor data or visual data for further processing, in the most existing designs, the issue of IoT security is always not taken into consideration. To ensure the information security, adopting cryptographic functions for data encryption is a common solution [31]–[33]. For a crop growth monitoring system, when the amount of sensor data becomes more and more, the decryption computation also incurs a heavy burden. Fortunately, FPGAs have also been demonstrated that they were suited to implement cryptographic functions [34]–[36] due to the advantages such as algorithm agility, resource efficiency, architecture efficiency and throughput. However, different sensor platforms and network environments usually use different cryptographic functions. This variety of cryptographic functions supported also affects the usability of the crop growth monitoring system directly. As a result, besides high-performance cryptographic computations, system scalability needs to be also taken into consideration.

Based on the above discussions, making a crop growth monitoring system smarter, securer and more scalable is urgently required. In this work, we use the FPGA device as the main computing architecture of the crop growth monitoring system. An edge AI based estimation method of PDS using the BNN and an adaptive cryptography engine are proposed to support smart system management. Details are introduced in Section III, while Table 1 gives the comparisons between our proposed crop growth monitoring system and the state-of-the-art designs.

III. PROPOSED CROP GROWTH MONITORING SYSTEM

To support high-performance CNN inference and cryptographic computing, in this work, the FPGA device is used to implement the proposed crop growth monitoring system. Our FPGA-based system architecture is given in Figure 2. An operating system (OS) runs on the microprocessor. The USB controller is used to interface with cameras and a wireless card. Here, sensor data are received through the wireless card, while the crop images are captured by the camera in real-time. A BNN hardware module is integrated into the FPGA-based system design to support the edge AI based estimation of PDS. Furthermore, the processor configuration access port (PCAP) and reconfigurable partitions (RPs),

TABLE 1. Comparisons on different designs.

Work (Ref.)	Sensor data		Visual data	
	Collect.	Security	CNN	Edge
Reynolds et al. [11]	√	×	×	×
Aravind and Maheswari [12]	√	×	×	×
Grimblatt et al. [13]	√	×	×	×
Ahmed et al. [14]	√	×	×	×
Lakshmi and Gayathri [15]	√	×	×	×
Nagasubramanian et al. [16]	√	×	√	×
Zhang et al. [18]	×	×	√	×
Thangadeepiga and Raja [19]	×	×	√	×
Sankar et al. [20]	×	×	√	×
Zhang et al. [21]	×	×	√	×
Yu et al. [22]	×	×	√	√
Lammie et al. [30]	×	×	√	√
Ours	√	√	√	√

Collect.: collection; Edge: edge computing; √: applicable; ×: not applicable.

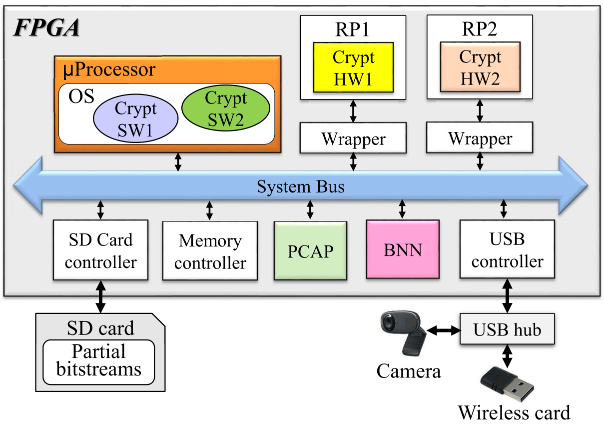

FIGURE 2. FPGA-based system architecture.

including RP1 and RP2, are used to realize the proposed adaptive cryptography engine. The RP1 and RP2 can be configured as the cryptographic hardware functions on-demand, while the wrappers are used to connect them to the system bus. Details are given in the following sections.

A. EFFICIENT ESTIMATION OF PDS WITH EDGE AI AND IMAGE FUSION

To estimate the PDS of the target crops in real-time, based on the discussions as described in Section II, a powerful embedded platform and a refined CNN architecture are both necessary. In this work, a resource-efficient BNN is presented to detect the target object, while an image fusion method is proposed to estimate the PDS.

1) DETECTION OF TARGET CROPS USING BNN

In this work, the VGG16 architecture [37] is adopted to support the detection of target crops. Similar to the inference engine [30], the VGG16 architecture is further refined through binary weight regularization [38] to support edge AI. As depicted in Equation 1, the weights and the activations of the BNN used in the crop growth monitoring system are constrained to either $+1$ or -1, while the activation function is applied to all BNN layers. Our BNN architecture is given in Figure 3. The CIFAR-10 dataset [39] is used as its input

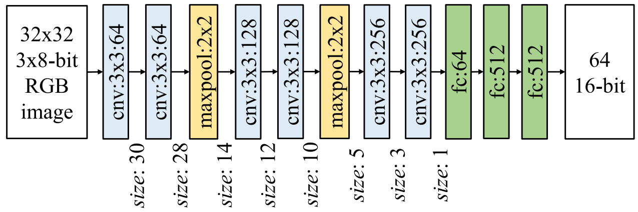

FIGURE 3. BNN architecture.

format, and the input of the BNN is a 32×32.8-bit RGB image. The BNN architecture can classify ten categories of objects, and it consists of six convolutional (cnv) operations, two maximum polling (maxpool) ones and three fully-connected (fc) ones. The kernel size of the maximum pooling operation and that of the convolutional operation are 2×2 and 3×3, respectively. The depths of the convolutional layers and the fully-connected ones of the BNN architecture are 64, 64, 128, 128, 256, 256, 64, 512 and 512 individually. Furthermore, the amount of all the parameters is 1.55 Mbits.

$$Sign(x) = \{+1 \ if \ x \geq 0, -1 \ if \ x < 0\} \tag{1}$$

Our FPGA-based system design flow is given in Figure 4. To quickly reflect the refinement of the BNN architecture and to make the corresponding hardware module easily replaceable without rewriting the corresponding hardware description language (HDL) codes, a framework called FINN [40] is used in this work. By integrating the deep learning framework with the FINN, after the training phase finishes, the BNN parameters can be exported and then used to generate the corresponding BNN hardware module. As a result, the high-level synthesizer (HLS) is further used to translate the high-level languages into the HDL codes. Finally, using the FPGA design tool, the BNN hardware module can be generated and integrated into our system design.

2) IMAGE FUSION METHOD FOR ESTIMATING THE PDS

Calculating the size of a discoloured area relative to the whole target crop is a conventional method to estimate the PDS [17]. However, it is easily affected by background objects. In order to enhance the estimation accuracy, besides RGB images, deep images captured by the ToF camera are also used in this work. Through the assistance of the agricultural expert, the PDS is divided into five levels. The increase in the value of level indicates the increase in the PDS. The proposed image fusion method for estimating the PDS is given in Algorithm 1.

The function CONTOUR is responsible for extracting all the positions of contours in a specific object (Lines 1-5), where *CTR* represents the set of these positions. IMG_{RGB} represents the RGB image in which the target crop can be recognized using the BNN, while IMG_{deep} represents the corresponding deep image. By considering the distance from the target crop to the ToF camera, the mask $mask_{crop}$ is used to extract the target crop from the deep image. Based on the characteristics of color, another mask $mask_{severity}$ is used to to extract the area of pests and diseases from the whole target crop.

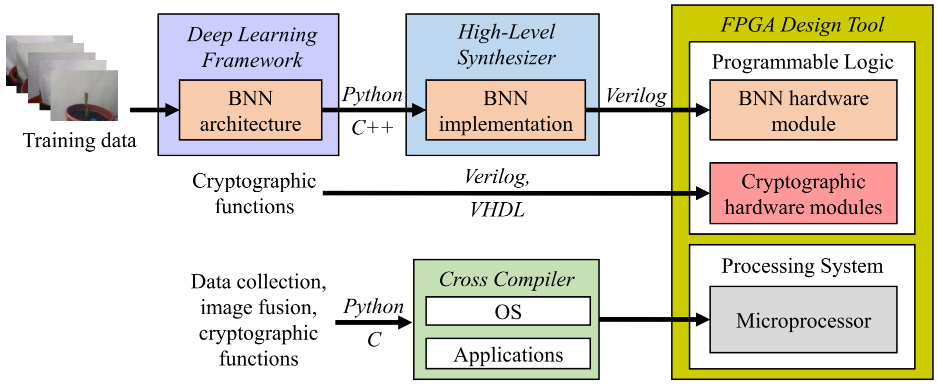

FIGURE 4. **FPGA-based design flow.**

Algorithm 1 Image Fusion Method for Estimating the PDS

1: **function** Contour(IMG, $mask$)
2: $IMG_{masking} \leftarrow masking(IMG, mask)$
3: $CTR \leftarrow findContours(IMG_{masking})$
4: **return** CTR
5: **end function**
6: $CTR_{crop} \leftarrow CONTOUR(IMG_{deep}, mask_{crop})$
7: $CNT_{crop} \leftarrow contourArea(CTR_{crop})$
8: $IMG_{onlycrop} \leftarrow extract(IMG_{RGB}, CTR_{crop})$
9: $CTR_{severity} \leftarrow CONTOUR(IMG_{onlycrop}, mask_{severity})$
10: $CNT_{severity} \leftarrow contourArea(CTR_{severity})$
11: $ratio \leftarrow \dfrac{CNT_{severity}}{CNT_{crop}}$
12: **if** $ratio \geq 80\%$ **then**
13: $Level \leftarrow 4$
14: **else if** $ratio \geq 60\%$ **then**
15: $Level \leftarrow 3$
16: **else if** $ratio \geq 40\%$ **then**
17: $Level \leftarrow 2$
18: **else if** $ratio \geq 20\%$ **then**
19: $Level \leftarrow 1$
20: **else**
21: $Level \leftarrow 0$
22: **end if**

In this image fusion method, based on the mask $mask_{crop}$, all the positions of contours in the target crop (CTR_{crop}) are first extracted, and then the pixel counts of the target crop (CNT_{crop}) are calculated (Lines 6-7). Next, all the positions of contours in the target crop (CTR_{crop}) is used to extract the area of the target crop from the RGB image (IMG_{RGB}) (Line 8). Using the predefined mask $mask_{severity}$ and the function `CONTOUR`, the area of pests and diseases is contoured ($CTR_{severity}$) from the image having only the target crop without the background ($IMG_{onlycrop}$). The pixel counts of area of pests and diseases ($CNT_{severity}$) are then calculated (Lines 9-10). Finally, similar to the estimation method [17], the ratio of $CNT_{severity}$ to CNT_{crop} is used to estimate which PDS level the target crop belongs to ($Level\ 0 \sim Level\ 4$). Here, based on the experience of agricultural experts, the five ratios of $CNT_{severity}$ to CNT_{crop} as shown in Algorithm 1 are used to represent the five levels of PDS. Then, according to the PDS level, the smart crop growth monitoring system can decide if the biological agents will be applied to the target crops to protect them from pests and diseases.

B. ADAPTIVE PROTECTION OF SENSOR DATA

In the proposed crop growth monitoring system, we design an adaptive cryptography engine as shown in Figure 1 to decrypt sensor data. Besides the cryptographic software functions, the corresponding hardware modules are also implemented and attached to the system bus for supporting real-time data decryption, as shown in Figure 2.

Different sensor platforms and network environments may require different cryptographic functions. Based on the basic service of secure socket layer (SSL), only one of the cryptographic functions will be used when the negotiation between the system and the sensor platform finishes. This means each cryptographic function is not performed all the time. To not only support real-time data decryption but also enhance system adaptivity, our previously proposed hardware virtualization technique [41] is integrated into the adaptive cryptography engine to satisfy the requirement of different cryptographic functions. As shown in Figure 2, two RPs, namely `RP1` and `RP2`, are implemented in the FPGA. The cryptographic hardware functions are integrated with a partial reconfigurable hardware task template (PR template) and then implemented as reconfigurable hardware modules configured in `RP1` and `RP2`. When a reconfigurable hardware module is being configured into `RP1` or `RP2`, the hardware designs in the other parts of the FPGA still run uninterrupted. Furthermore, an SD card is used to store all the partial bitstreams corresponding to the RPs, which are configured in the FPGA through the PCAP.

According to the FPGA-based design as shown in Figure 2, a layered and virtualizable system design is introduced in this work, as shown in Figure 5. The hardware design is divided into the logical hardware layer and the physical one. According to different requirements, the adaptive cryptography engine enables `RP1` and `RP2` to be configured as the

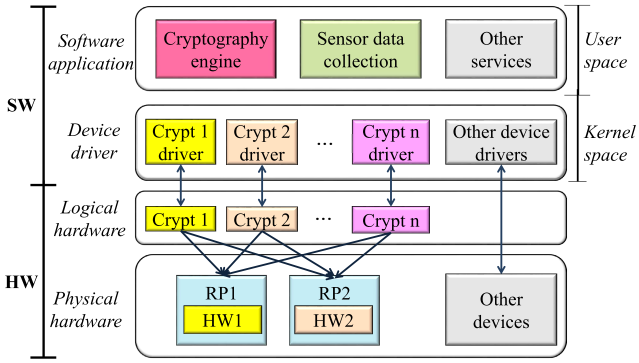

FIGURE 5. Layered and virtualizable system design.

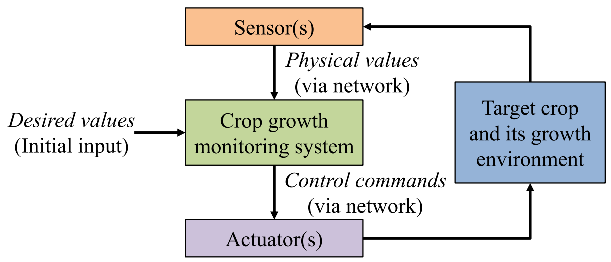

FIGURE 6. Closed-loop control model.

requested cryptographic function at runtime. As a result, the RP can be virtualized as requested hardware devices in the logical hardware layer on demand, even though the total logic resource requirement of all hardware modules exceeds the amount of resources available in an FPGA device. Therefore, software applications can access these hardware devices in the logical hardware layer through their corresponding device drivers.

C. AGRICULTURAL CYBER PHYSICAL SYSTEM

To monitor the growth status of the target crops more efficiently, a smart system management mechanism, including a hardware/software adaptation mechanism and a crop growth management mechanism, is also proposed in this work. Details are given in the following sections.

1) CROP GROWTH MANAGEMENT

According to the PDS and the water need of crops, the crop growth monitoring system can perform the corresponding actions to interact with the physical world to ensure the healthy growth of crops. As shown in Figure 6, a closed-loop control model is further presented in this work.

- Actions based on the PDS: According to the experiences of agricultural experts, the PDS level must be less than `Level 2`. By using the physical values captured by the RGB camera and the ToF one, the crop growth monitoring system performs the image fusion method for estimating the PDS as shown in Algorithm 1. When the PDS level is estimated more than `Level 2`, a nozzle is used to apply the biological agents to the target crops to protect them from pests and diseases.
- Actions based on the water need: The root zone soil moisture deficit (RZSMD) [42] is used in the crop growth monitoring system. Its objective is to make RZSMD close to zero, while the amount of irrigation water use can be minimized. Given D as RZSMD at the current time step, RZSMD at the next time step is D^+ as depicted in Equation 2.

$$D^+ = D + E^* - P^e - I^e \tag{2}$$

Here, E^* is the crop evapotranspiration, which is calculated based on the RZSMD approach [42] by measuring the soil moisture of the target crop. P^e is the effective rainfall. Since our target crops are planted in the greenhouse, the value of P^e is set as zero. I^e is the effective irrigation amount, which is controlled through the sprinklers. Therefore, the crop growth monitoring system will make RZSMD (D^+) close to zero by requesting the sprinklers to irrigate the target crops. Note that the RZSMD approach [42] depends heavily on the conditions of the soil. When another different type of soil is used, the crop evapotranspiration needs to be recalculated.

2) HARDWARE/SOFTWARE ADAPTATION

Based on the adaptive cryptography engine as introduced in Section III-B, the proposed hardware/software adaptation mechanism is as illustrated in Figure 7. When the crop growth monitoring system receives a request for receiving sensor data from a sensor platform, it then negotiates with the sensor platform to use the same cryptographic function. Otherwise, when no request is received for a constant time, the blank modules will be configured in `RP1` and `RP2`, as shown in Figure 2, to reduce the power consumption.

When the crop growth monitoring system finishes the negotiation with the sensor platform, it checks if the requested cryptographic function has been configured in the RPs. If the requested cryptographic function has been configured in the RPs and is not being used by a software application, the crop growth monitoring system will start to receive and decrypt the sensor data. Otherwise, if the requested cryptographic function is not configured in the RPs or it has been configured but is being used by a software application, the crop growth monitoring system will check if there is still an idle RP to configure the requested cryptographic function. When there is an idle RP, the requested cryptographic function is thus configured in the idle RP; otherwise, the corresponding software version will be used.

IV. SYSTEM IMPLEMENTATION AND EVALUATION

To demonstrate the practicability of the proposed crop growth monitoring system, this section will introduce our system implementation and evaluation.

A. SYSTEM IMPLEMENTATION

To realize the smart crop growth monitoring in the real planting environment, edge AI is integrated into our system design.

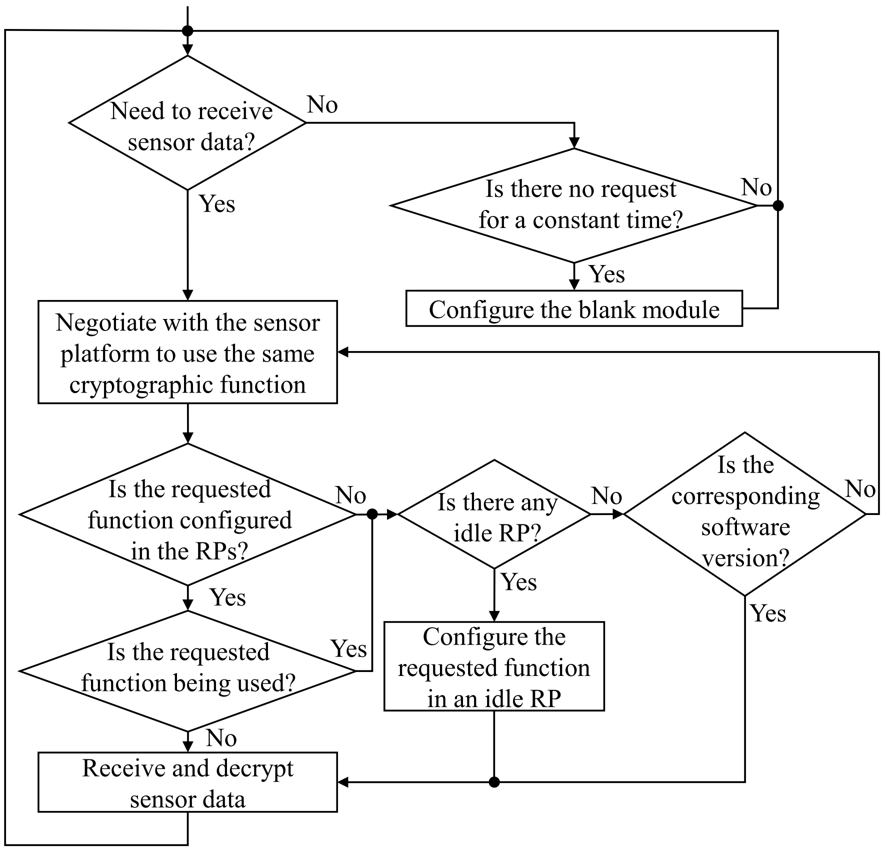

FIGURE 7. Hardware/software adaptation.

Instead of adopting the GPU platform such as NVIDIA Jetson TX2 [22] for AI inference, the FPGA device was used in this work for the acceleration of the BNN, similar to the design [30] that adopted the Intel DE1-SoC development board. In our current implementation, the PYNQ-Z2 development board with a Xilinx Zynq-7000 XC7Z020 chip was used to implement the smart crop growth monitoring system. Furthermore, dragon fruits were adopted as the target crops.

Figure 8 shows our system setup. Here, sensors were attached to the Arduino WeMos D1 boards, where sensor data were transferred to the smart crop growth monitoring system (PYNQ-Z2) via WIFI. A Logitech C922 Pro stream camera and an ADI ToF module were attached to the smart crop growth monitoring system (PYNQ-Z2) for capturing the RGB images and deep ones, respectively. Furthermore, besides a manual mode and an automatic mode for the control of greenhouse equipment, the greenhouse control system was customized to provide an external control interface based on an Arduino platform. For the control of the greenhouse equipment such as sprinklers, the smart crop growth monitoring system (PYNQ-Z2) can transfer the control command to the external control interface of the greenhouse control system via an UART interface. Therefore, applying the smart crop growth monitoring system to agricultural automation can be achieved. Note that as shown in Figure 8 the smart crop growth monitoring system was mainly implemented using the PYNQ-Z2 development board quipped with a Logitech

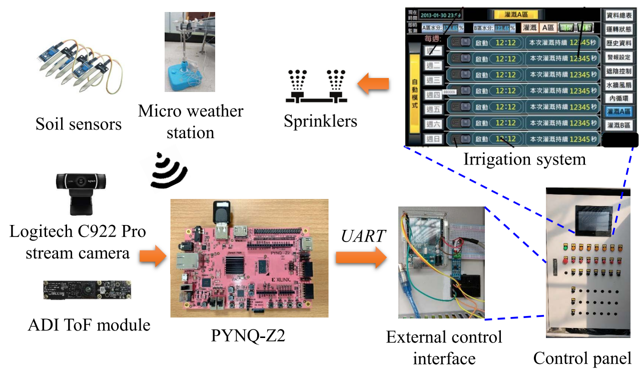

FIGURE 8. System setup for agricultural automation.

C922 Pro stream camera, an ADI ToF module and a wireless USB adapter. The system cost was around NT$12,000. Furthermore, the soil sensors were used to measure the soil parameters such as pH, moisture, salinity and temperature, while the micro weather station was used to measure the environment parameters such barometric pressure, temperature and humidity. Here, all the sensors and the corresponding platforms cost around NT$50,000.

B. BNN DATASET

Our environmental setup of data collection in the greenhouse is shown in Figure 9. One hundred dragon fruits were planted in a greenhouse. Similar to the estimation method [17] by

TABLE 2. Required time for the estimation of PDS.

Operation	Required time (ms)
Image capture (t_{cap})	38.6
BNN detection (t_{BNN})	768.1
Classification (t_{cla})	15,656.6
Total (T)	16,463.3

calculating the size of a deformed or discoloured area relative to the whole crop, the PDS for each dragon fruit was classified through the assistance of the agricultural expert every day as shown in Figure 10. Furthermore, for each level of PDS, the corresponding dragon fruit on the display turntable was also exchanged every day to ensure the data diversity. Based on such a data collection design, five thousand images of the target crops and forty-five thousand images having nine categories of objects (five thousand images per category) were used as our training data. Here, the nine categories consisted of acorn, banana, bell pepper, cauliflower, spider, ladybug, lemon, mushroom, and orange, while their corresponding images were obtained the ImageNet dataset [43]. Furthermore, all the images used as training data were resized as 32 × 32 images.

1) PDS ESTIMATOR

Based on the FPGA-based design flow as illustrated in Figure 4, Theano [44] was adopted to train the BNN, and it was executed on a server (Intel Core i7-8700 CPU, 64 GB RAM and Nvidia GeForce RTX 2080 GPU). The number of training epochs was set as 500, and the batch normalization [45] was applied to the BNN training for acceleration. The FINN [40] was adopted to generate the corresponding BNN hardware module through the parameters obtained from the BNN training. Since the generated BNN module was based on a Xilinx Vivado HLS project, the compiled HDL results were extracted and integrated into the crop growth monitoring system. Furthermore, as shown in Figure 9, soil sensors were used to measure the soil moisture for the estimation of crop evapotranspiration.

2) ADAPTIVE CRYPTOGRAPHY ENGINE

In our current implementation, the crop growth monitoring system can support four cryptographic functions, namely repaired tiny encryption algorithm 32 (RTEA32), RTEA64, eXtended TEA 32 (XTEA32) and XTEA64, by implementing two PRs, namely `RP1` and `RP2`. Note that the value embedded in the name of each cryptographic function represents its input data sizes in bits. The hardware intellectual properties (IPs) corresponding to the four cryptographic functions were obtained from the OpenCores website. Each cryptographic IP was integrated with the PR template [41] and then implemented as two reconfigurable hardware modules configured in `RP1` and `RP2`, respectively. Note that the current implementation is only a proof-of-concept. A different FPGA chip having more gate counts can be also used to integrate with newer and securer cryptographic functions to ensure the security of sensor data.

C. EVALUATION OF PDS ESTIMATOR

In the following, we will evaluate the PDS estimator in terms of performance and recognition accuracy.

1) PERFORMANCE

Based on Algorithm 1, the RGB camera and the ToF one are used to estimate the PDS of the target crops i.e. dragon fruits. A real case is shown in Figure 11. Given the time of t_{cap} milliseconds (ms) to capture the image data, the time of t_{BNN} ms to detect the target crop using the BNN, the time of t_{cla} ms to classify the PDS level of the target crop, the total time of *Total* ms is as depicted in Equation 3. Here, n_{reg} represents the number of regions extracted from the captured image using the selective search method [46], while n_{tar} represents the number of regions that the BNN can detect the target crop.

$$Total = t_{cap} + n_{reg} \times t_{BNN} + n_{tar} \times t_{cla} \tag{3}$$

The time required by each operation in our current implementation is given in Table 2. Here, the detection time using BNN (t_{BNN}) includes both the processing time by using the BNN hardware module and the time to deliver a region to the BNN hardware module. Experimental results show that estimating the PDS of the target crop ($n_{reg} = 1$ and $n_{tar} = 1$) requires $16,463.3$ ms. However, the resolution of the captured image is based on Full HD. Through the selective search method [46], many regions in the captured image will be extracted and resized for the detection of the target crop using the BNN. This means the BNN detection will be performed many times, and system performance will be heavily affected by the time to perform the BNN. To further evaluate the PDS estimator of the proposed crop growth monitoring system, the same BNN architecture as shown in Figure 3 was also implemented on the microprocessor-based design, that is, ARM Cortex A9, and the GPU accelerated one, that is, NVIDIA GeForce RTX 2080, for comparison. The experimental results are given in Table 3.

Compared to the microprocessor-based design and the GPU accelerated one, when the same BNN architecture was performed on the FPGA, the frames per second (FPS) can be accelerated by a factor of 4,919.29 and a factor of 1.08, respectively. According to the experiments, the PDS estimator has been showed it can provide better performance to support the real-time detection of the target crops at the edge. Such an advantage is very important especially for the crop growth monitoring. Note that, to perform the same BNN model on GPU, in this experiment, the NVIDIA GeForce RTX 2080 GPU was adopted directly used for comparison. Although the optimization method such as the NIVIDIA TensorRT was not used, this comparison is also unfair for our FPGA-based implementation. This is because the NVIDIA GeForce RTX 2080 GPU is usually used in a server due to the capability of powerful computation. Furthermore, the crop growth monitoring system needs to be applied to the real scene, but the NVIDIA GeForce RTX 2080 GPU is not designed for the edge device. To perform the same BNN model, it can be easily expected our

FIGURE 9. Data collection in the greenhouse.

TABLE 3. BNN implemented on different computing architectures.

Computing architectures	FPS
μprocessor-based (ARM Cortex-A9)	0.62
GPU accelerated (NVIDIA GeForce RTX 2080)	2,815.11
Ours (Xilinx Zynq-7000 XC7Z020)	3,049.96

proposed FPGA-based design can save much more energies compared to the NVIDIA GeForce RTX 2080 GPU. Thus, the experiments on energy were not given in this work. This experiment is mainly used to demonstrate the performance using our FPGA-based design on the PYNQ-Z2 platform to perform the BNN inference can be close to or even higher than that using the GPU acceleration on the NVIDIA GeForce RTX 2080. This also indicates at the edge, compared to the embedded GPU platform such as NVIDIA Jetson Nano, the FPGA-based design could be a considerable choice. This work can be also considered as a case study of FPGA inference at the edge.

2) RECOGNITION ACCURACY

To evaluate the recognition accuracy of the BNN hardware module, ten thousand images containing one thousand images of the target crops and the nine categories of objects (one thousand images per category) as described in Section IV-A were used for testing. Our experiment shows the recognition accuracy of target crops by using the BNN hardware module of the crop growth monitoring system can achieve 76.57%. However, as introduced in Algorithm 1, in the real scene as shown in Figure 9, besides the target crops i.e. dragon fruits, most of the remaining objects will be filtered out through the masks $mask_{crop}$ and $mask_{severity}$ based on deep characteristics and colors, respectively. Therefore, the recognition accuracy (76.57%) in our real scene can be accepted. Furthermore, the classification method of the five PDS levels in Algorithm 1 was based on the experience of agricultural experts, which mainly depended on the size of a discoloured area relative to the whole crop. By using Algorithm 1, the target crop could be extracted by removing the background objects. Thus, the results of PDS classification using the proposed system could be close to those classified by agricultural experts.

FIGURE 10. Five levels of PDS.

D. EVALUATION OF ADAPTIVE CRYPTOGRAPHY ENGINE

In the following, we will evaluate the adaptive cryptography engine in terms of performance, resource usages and power consumption.

1) PERFORMANCE

In this experiment, the RTEA32, RTEA64, XTEA32 and XTEA64 functions were also implemented as software programs executed on the ARM cortex A9 processor (667MHz) embedded in the PYNQ-Z2 development board for comparison. The same amount of operations (100 decryption operations) were performed by using the RTEA32, RTEA64, XTEA32 and XTEA64 hardware modules and their corresponding software designs. Figure 4 gives the processing time required by the four cryptographic functions. Experiments show that the processing time using the hardware modules can be reduced by 39.4% to 80.1%, compared to the corresponding software designs. Therefore, as shown in Figure 7, the hardware module of the requested cryptographic function will be first considered so that the possibility of real-time data decryption can be increased.

2) RESOURCE USAGES

Another system design without the adaptive cryptography engine was also implemented for comparison.

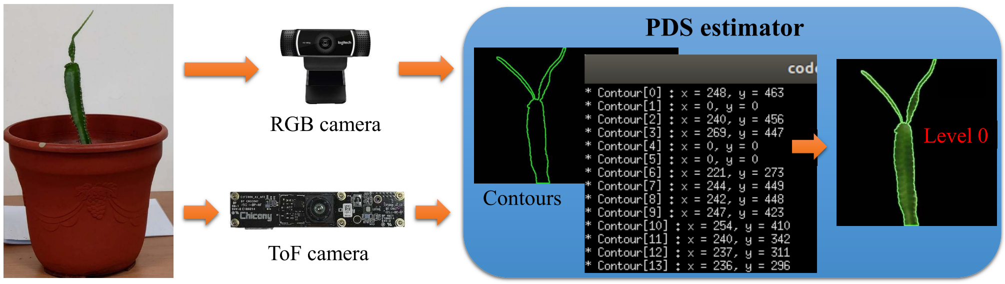

FIGURE 11. A real case using the image fusion method for estimating the PDS.

TABLE 4. Processing time (μs) required by the four cryptographic functions.

Function	Reconfigurable hardware	Software program
XTEA64	1,891	3,123
XTEA128	1,895	3,210
RTEA64	614	3,036
RTEA128	619	3.124

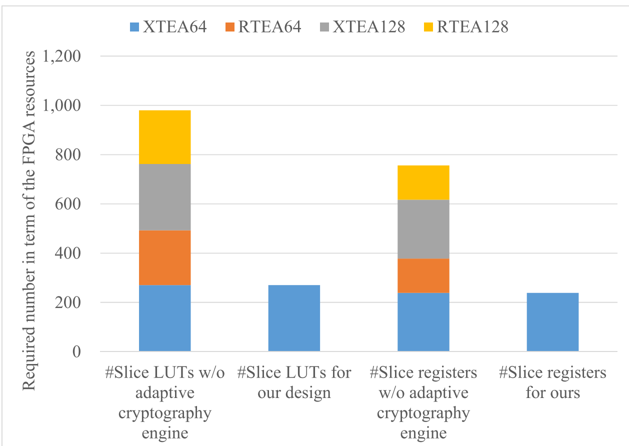

FIGURE 12. Resource usages.

For a traditional system design, to support all the four cryptographic functions, namely RTEA32, RTEA64, XTEA32 and XTEA64, the corresponding hardware modules need to be first configured in the system at design-time. When the negotiation between the system and the sensor platform finishes, only one of the cryptographic hardware modules will be used. This indicates that the four cryptographic hardware modules are not always performed, which also results in a waste of system resources. By using the adaptive cryptography engine, each RP can be configured as the requested cryptographic module on-demand. To support all the four cryptographic functions, Figure 12 shows the resource usages in terms of slice LUTs and slice registers when the adaptive cryptography engine was used (our design) or not.

When the adaptive cryptography engine was used, the crop growth monitoring system needed at most 540 slice LUTs and 478 slice registers, considering the Xilinx XC7Z020 chip. This shows the maximal resource usage by the XTEA64 hardware modules configured in both `RR1` and `RP2`. Compared to the system design without the adaptive cryptography engine,

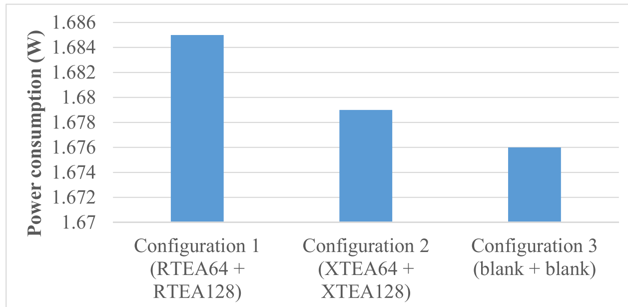

FIGURE 13. Power consumption.

our design can reduce 72.4% of slice LUTs and 68.4% of slice registers. Furthermore, our crop growth monitoring system design is scalable due to its layered and virtualizable design. New hardware functions can be also integrated into the system. This means that the crop growth monitoring system can support much more hardware modules than our current implementation can.

3) POWER CONSUMPTION

As illustrated in Figure 7, when no request is received for a constant time, the blank modules corresponding to the two RPs can be configured in the system to adapt the power consumption. In this experiment, the Xilinx Vivado 2018.2 tool was used to measure the power consumption of the placed and routed netlists. For different combinations of cryptographic hardware modules, Figure 13 gives the measured power consumptions in watt (W). Compared to the worst case of using maximum power for each of the two RPs (`Configuration 1`), that is, the RTEA64 function in `RP1` and the RTEA128 function in `RP2`, the total power consumption can be reduced by 0.009 watts when the corresponding blank modules (`Configuration 3`) were configured in `RP1` and `RP2`. Therefore, the power consumption of the crop growth monitoring system can be dynamically adapted at runtime, according to varying cryptographic functions.

V. CONCLUSION

To monitor the crop growth efficiently, this work proposes a crop growth monitoring system based on system adaptivity and edge AI. The presented adaptive cryptography engine can not only support varying requirements of cryptographic functions but also provide real-time decryption processing of sensor data. Furthermore, the layered and virtualizable design makes the crop growth monitoring system scalable. The edge AI based PDS estimator provides real-time detection of the target crops, while the image fusion method can assist in classifying the level of PDS. Through the smart system management mechanism along with the adaptive cryptography engine and the PDS estimator, the actuators can be controlled to interact with the physical world to ensure the healthy growth of crops. Our experiments also demonstrated the practicability and applicability of the proposed design.

REFERENCES

[1] C.-R. Rad, O. Hancu, I.-A. Takacs, and G. Olteanu, "Smart monitoring of potato crop: A cyber-physical system architecture model in the field of precision agriculture," *Agricult. Agricult. Sci. Proc.*, vol. 6, pp. 73–79, Sep. 2015.

[2] R. Rathna, U. V. Anbazhagu, L. Mary Gladence, V. M. Anu, and J. Sybi Cynthia, "An intelligent monitoring system for water quality management using Internet of Things," in *Proc. 8th Int. Conf. Smart Comput. Commun. (ICSCC)*, Jul. 2021, pp. 291–297.

[3] V. R. Kamala and L. MaryGladence, "An optimal approach for social data analysis in big data," in *Proc. Int. Conf. Comput. Power, Energy, Inf. Commun. (ICCPEIC)*, Apr. 2015, pp. 0205–0208.

[4] L. M. Gladence, K. R. Reddy, M. P. Reddy, and M. P. Selvan, "A prediction of crop yield using machine learning algorithm," in *Proc. 5th Int. Conf. Trends Electron. Informat. (ICOEI)*, Jun. 2021, pp. 1072–1077.

[5] Z.-Q. Zhao, P. Zheng, S.-T. Xu, and X. Wu, "Object detection with deep learning: A review," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 11, pp. 3212–3232, Nov. 2019.

[6] S. Liang, S. Yin, L. Liu, W. Luk, and S. Wei, "FP-BNN: Binarized neural network on FPGA," *Neurocomputing*, vol. 275, pp. 1072–1086, Jan. 2018.

[7] J. Qiu, J. Wang, S. Yao, K. Guo, B. Li, E. Zhou, J. Yu, T. Tang, N. Xu, S. Song, Y. Wang, and H. Yang, "Going deeper with embedded FPGA platform for convolutional neural network," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays*, Feb. 2016, pp. 26–35.

[8] V. Hassija, V. Chamola, V. Saxena, D. Jain, P. Goyal, and B. Sikdar, "A survey on IoT security: Application areas, security threats, and solution architectures," *IEEE Access*, vol. 7, pp. 82721–82743, 2019.

[9] W. Iqbal, H. Abbas, M. Daneshmand, B. Rauf, and Y. A. Bangash, "An in-depth analysis of IoT security requirements, challenges, and their countermeasures via software-defined security," *IEEE Internet Things J.*, vol. 7, no. 10, pp. 10250–10276, Oct. 2020.

[10] J. Kar, K. Naik, and T. Abdelkader, "A secure and lightweight protocol for message authentication in wireless sensor networks," *IEEE Syst. J.*, vol. 15, no. 3, pp. 3808–3819, Sep. 2021.

[11] D. Reynolds, J. Ball, A. Bauer, R. Davey, S. Griffiths, and J. Zhou, "CropSight: A scalable and open-source information management system for distributed plant phenotyping and IoT-based crop management," *GigaScience*, vol. 8, no. 3, Jan. 2019, Art. no. giz009.

[12] R. Aravind and D. Maheswari, "Plant growth monitoring system," *Int. J. Sci. Technol. Res.*, vol. 9, no. 4, pp. 482–487, Apr. 2020.

[13] V. Grimblatt, C. Jégo, G. Ferré, and F. Rivet, "How to feed a growing population—An IoT approach to crop health and growth," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 11, no. 3, pp. 435–448, Sep. 2021.

[14] N. Ahmed, D. De, and I. Hussain, "Internet of Things (IoT) for smart precision agriculture and farming in rural areas," *IEEE Internet Things J.*, vol. 5, no. 6, pp. 4890–4899, Dec. 2018.

[15] K. Lakshmi and S. Gayathri, "Implementation of IoT with image processing in plant growth monitoring system," *J. Sci. Innov. Res.*, vol. 6, no. 2, pp. 80–83, 2017.

[16] G. Nagasubramanian, R. K. Sakthivel, R. Patan, M. Sankayya, M. Daneshmand, and A. H. Gandomi, "Ensemble classification and IoT-based pattern recognition for crop disease monitoring system," *IEEE Internet Things J.*, vol. 8, no. 16, pp. 12847–12854, Aug. 2021.

[17] N. Winstead and A. Kelman, "Inoculation techniques for evaluating resistance to pseudomonas solanacearum," *Phytopath*, vol. 42, no. 11, pp. 628–634, 1952.

[18] L. Zhang, Z. Xu, D. Xu, J. Ma, Y. Chen, and Z. Fu, "Growth monitoring of greenhouse lettuce based on a convolutional neural network," *Horticulture Res.*, vol. 7, no. 1, p. 124, Aug. 2020.

[19] E. Thangadeepiga and R. A. A. Raja, "Remote sensing-based crop identification using deep learning," in *Intelligent Data Engineering and Analytics* (Advances in Intelligent Systems and Computing), vol. 1177, S. Satapathy, Y. D. Zhang, V. Bhateja, and R. Majhi, Eds. Singapore: Springer, doi: 10.1007/978-981-15-5679-1_11.

[20] P. R. Sai Sankar, S. RamaKrishna, M. M. Venkata Rakesh, P. Raja, V. T. Hoang, and C. Szczepanski, "Intelligent health assessment system for paddy crop using CNN," in *Proc. 3rd Int. Conf. Signal Process. Commun. (ICPSC)*, May 2021, pp. 382–387.

[21] L. Zhang, G. Gui, A. M. Khattak, M. Wang, W. Gao, and J. Jia, "Multi-task cascaded convolutional networks based intelligent fruit detection for designing automated robot," *IEEE Access*, vol. 7, pp. 56028–56038, 2019.

[22] Y. Yu, K. Zhang, H. Liu, L. Yang, and D. Zhang, "Real-time visual localization of the picking points for a ridge-planting strawberry harvesting robot," *IEEE Access*, vol. 8, pp. 116556–116568, 2020.

[23] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Quantized neural networks: Training neural networks with low precision weights and activations," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 6869–6898, 2017.

[24] N. J. Fraser, Y. Umuroglu, G. Gambardella, M. Blott, P. Leong, M. Jahre, and K. Vissers, "Scaling binarized neural networks on reconfigurable logic," in *Proc. 8th Workshop 6th Workshop Parallel Program. Run-Time Manage. Techn. Many-Core Archit. Design Tools Archit. Multicore Embedded Comput. Platforms*, Jun. 2017, pp. 25–30.

[25] A. Shawahna, S. M. Sadiq, and A. Ei-Maleh, "FPGA-based accelerators of deep learning networks for learning and classification: A review," *IEEE Access*, vol. 7, pp. 7823–7859, 2018.

[26] E. Nurvitadhi, D. Sheffield, J. Sim, A. Mishra, G. Venkatesh, and D. Marr, "Accelerating binarized neural networks: Comparison of FPGA, CPU, GPU, and ASIC," in *Proc. Int. Conf. Field-Program. Technol. (FPT)*, Dec. 2016, pp. 77–84.

[27] R. Zhao, W. Song, W. Zhang, T. Xing, J.-H. Lin, M. Srivastava, R. Gupta, and Z. Zhang, "Accelerating binarized convolutional neural networks with software-programmable FPGAs," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays*, Feb. 2017, pp. 15–24.

[28] D. J. M. Moss, E. Nurvitadhi, J. Sim, A. Mishra, D. Marr, S. Subhaschandra, and P. H. W. Leong, "High performance binary neural networks on the Xeon+FPGA platform," in *Proc. 27th Int. Conf. Field Program. Logic Appl. (FPL)*, Sep. 2017, pp. 1–4.

[29] P. Jokic, S. Emery, and L. Benini, "BinaryEye: A 20 kfps streaming camera system on FPGA with real-time on-device image recognition using binary neural networks," in *Proc. IEEE 13th Int. Symp. Ind. Embedded Syst. (SIES)*, Jun. 2018, pp. 1–17.

[30] C. Lammie, A. Olsen, T. Carrick, and M. R. Azghadi, "Low-power and high-speed deep FPGA inference engines for weed classification at the edge," *IEEE Access*, vol. 7, pp. 51171–51184, 2019.

[31] P.-Y. Ting, J.-L. Tsai, and T.-S. Wu, "Signcryption method suitable for low-power IoT devices in a wireless sensor network," *IEEE Syst. J.*, vol. 12, no. 3, pp. 2385–2394, Sep. 2018.

[32] S. S. Dhanda, B. Singh, and P. Jindal, "Lightweight cryptography: A solution to secure IoT," *Wireless Pers. Commun.*, vol. 112, no. 3, pp. 1947–1980, Jan. 2020.

[33] V. A. Thakor, M. A. Razzaque, and M. R. Khandaker, "Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities," *IEEE Access*, vol. 9, pp. 28177–28193, 2021.

[34] I. Algredo-Badillo, C. Feregrino-Uribe, R. Cumplido, and M. Morales-Sandoval, "Efficient hardware architecture for the AES-CCM protocol of the IEEE 802.11I standard," *Comput. Electr. Eng.*, vol. 36, no. 3, pp. 565–577, May 2010.

[35] A. P. Fournaris and N. Sklavos, "Secure embedded system hardware design—A flexible security and trust enhanced approach," *Comput. Elect. Eng.*, vol. 40, no. 1, pp. 121–133, Jan. 2014.

[36] T. Wollinger and C. Paar, "How secure are FPGAs in cryptographic applications," in *Proc. 13th IEEE Int. Conf. Field Program. Log. Appl.*, Aug. 2003, pp. 1–3.

[37] Z. Liu, J. Wu, L. Fu, Y. Majeed, Y. Feng, R. Li, and Y. Cui, "Improved kiwifruit detection using pre-trained VGG16 with RGB and NIR information fusion," *IEEE Access*, vol. 8, pp. 2327–2336, 2020.

[38] I. Hubara, M. Courbariaux, D. Soudry, R. El-Yaniv, and Y. Bengio, "Binarized neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2016, pp. 4107–4115.

[39] A. Krizhevsky, "Learning multiple layers of features from tiny images," Univ. Toronto, Toronto, ON, Canada, Tech. Rep., 2009.

[40] Y. Umuroglu, N. J. Fraser, G. Gambardella, M. Blott, P. Leong, M. Jahre, and K. Vissers, "FINN: A framework for fast, scalable binarized neural network inference," in *Proc. ACM/SIGDA Int. Symp. Field-Program. Gate Arrays*, Feb. 2017, pp. 65–74.
[41] C.-H. Huang and P.-A. Hsiung, "Virtualizable hardware/software design infrastructure for dynamically partially reconfigurable systems," *ACM Trans. Reconfigurable Technol. Syst.*, vol. 6, no. 2, pp. 1–18, Jul. 2013.
[42] D. Delgoda, H. Malano, S. K. Saleem, and M. N. Halgamuge, "Irrigation control based on model predictive control (MPC): Formulation of theory and validation using weather forecast data and AQUACROP model," *Environ. Model. Softw.*, vol. 78, pp. 40–53, Apr. 2016.
[43] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[44] J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-farley, and Y. Bengio, "Theano: A CPU and GPU math compiler in Python," in *Proc. 9th Python Sci. Conf.*, 2010, pp. 3–10.
[45] J. Bjorck, C. Gomes, B. Selman, and K. Q. Weinberger, "Understanding batch normalization," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, Dec. 2018, pp. 7705–7716.
[46] P. F. Felzenszwalb and D. P. Huttenlocher, "Efficient graph-based image segmentation," *Int. J. Comput. Vis.*, vol. 59, no. 2, pp. 167–181, Sep. 2004.

CHUN-HSIAN HUANG (Member, IEEE) received the Ph.D. degree in computer science and information engineering from National Chung Cheng University, Taiwan, in January 2011. He was a Post-Doctoral Scholar at the Intel-NTU Connected Context Computing Center, National Taiwan University, in July 2011. From August 2011 to February 2012, he was an Assistant Researcher at the Chung-Shan Institute of Science and Technology, Taiwan. Since February 2012, he has been a faculty with the Department of Computer Science and Information Engineering, National Taitung University, where he is currently a Professor. His research interests include embedded systems, reconfigurable computing, cyber physical systems, and robotic applications. Details can be found on the website. https://sites.google.com/site/chunhsianhuang/english-version

BO-WEI CHEN received the B.S. degree in computer science and information engineering from National Taitung University, Taiwan, in 2021. His research interests include embedded system design and artificial intelligence applications.

YI-JIE LIN received the B.S. degree in computer science and information engineering from National Taitung University, Taiwan, in 2021. Her research interests include embedded system design and artificial intelligence applications.

JIA-XUAN ZHENG received the B.S. degree in computer science and information engineering from National Taitung University, Taiwan, in 2021. Her research interests include embedded system design and artificial intelligence applications.