IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received August 20, 2021, accepted September 4, 2021, date of publication September 7, 2021,
date of current version September 15, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3110978

Multi-Scale Context Aggregation for Strawberry Fruit Recognition and Disease Phenotyping

TALHA ILYAS[1,2], ABBAS KHAN[1,2], MUHAMMAD UMRAIZ[1,2], YONGCHAE JEONG[3], (Senior Member, IEEE), AND HYONGSUK KIM[1,2], (Senior Member, IEEE)

[1]Division of Electronics and Information Engineering, Jeonbuk National University, Jeonju 54896, Republic of Korea
[2]Core Research Institute of Intelligent Robots, Jeonbuk National University, Jeonju 54896, Republic of Korea
[3]Division of Electronics and Information Engineering, IT Convergence Research Center, Jeonbuk National University, Jeonju-si 54896, South Korea

Corresponding authors: Yongchae Jeong (ycjeong@jbnu.ac.kr) and Hyongsuk Kim (hskim@jbnu.ac.kr)

This work was supported in part by the National Research Foundation of Korea (NRF) Grant through Korean Government under Grant NRF2019R1A2C1011297 and Grant NRF-2019R1A6A1A09031717, and in part by the Crop and Weed Project through the Agricultural Science and Technology Development Cooperation Research Program under Grant PJ01572002.

ABSTRACT Timely harvesting and disease identification of strawberry fruits is a major concern for commercial level cultivators. Failing to harvest the grown strawberries can result in the fruit rotting which makes their damaged tissues more prone to grey mold pathogens. Immediate removal of the overgrown or diseased strawberries is inevitable to curb the mass spreading of the pathogen. In this paper, we propose a deep learning-based framework to identify three different strawberry fruit classes (unripe, partially ripe and ripe), as well as a class of overgrown or diseased strawberries. We equip the proposed convolutional encoder-decoder network with three different modules. One for adaptively controlling receptive filed size of the network to detect objects of multiple sizes. Second for controlling the flow of salient features (information) to the deeper layers of the network and the other for controlling the architecture's computational complexity. These modules combined, outperform the previous state-of-the-art semantic segmentation networks on the task of strawberry fruit phenotyping. We also introduce a dataset collected from different farms to evaluate the performance of the network. Quantitative and qualitative results show that notwithstanding heterogeneity in the data and the effect of the real-field variations, our approach produced remarkable results with a 3% increase in mean intersection over union as compared to the other state-of-the-art networks and was able to recognize diseased fruits with a precision of 92.45%.

INDEX TERMS Deep learning, strawberries fruit recognition, segmentation, classification, disease phenotyping, smart farming, precision agriculture.

I. INTRODUCTION

Smart farming is a recently coined terminology to solve the problems in agriculture, related to production, environmental impact, and sustainability. With an increase in the global population, food demand is growing monotonically. The goal now is to produce plentiful quality nutritious food in a timely manner while at the same time protecting our ecosystem. To overcome these challenges, it is necessary to understand the complex and unpredictable agricultural ecosphere. Machine vision systems are therefore gradually being adopted in smart farming to automate agricultural tasks: from sowing to harvesting, with a minimum cost of production. The purpose is to raise the quality products by maintaining healthy environmental conditions. Strawberry is one of the major crops of Korea and the country exports a part of its production. Now with the advancements of the technology, labor-intensive tasks are being replaced by automated solutions as the young generation has less interest in agriculture labor. Therefore, the available manpower is insufficient with the high labor cost. Moreover, harvesting of the strawberry is also time taking and burdensome task since strawberries have no specific time to grow and they need to be harvested as soon as they are ripe. In the meantime, strawberries need to be continuously checked to see if they are ripe or not unless they will overgrow and start to rot.

Botrytis fruit rot (Gray Mold) is a common disease found in Strawberries. It affects not only the fruits but also

The associate editor coordinating the review of this manuscript and approving it for publication was Kumaradevan Punithakumar.

flower stalks, petals, and crowns of the strawberries and therefore causing a huge loss to the commercial production economically. In symptoms of this disease, there appears velvety grayish fungus on the fruits, which slowly covers the whole fruit and mummify it. Biologists claim that under wet conditions if not sprayed with fungicides, there could be an 80% loss of fruits and flowers [1]. Damaged/senescing tissues and those which deteriorate with age create a favorable environment for the gray mold. Therefore, overgrown strawberries that were not harvested on time are prone to this pathogen easily [2]. These old and overgrown strawberries infected with gray mold are responsible for transferring fungi to the healthy partially ripe and full ripe strawberries.

In the current decade, deep learning has revolutionized the artificial intelligence (AI) realm and continues to do so. Deep learning algorithms have shown remarkable performance practically, especially in computer vision [3]–[6]. In this context, machine vision approaches are a hot research area where robotic solutions are developed to automate the processes [7]–[13]. In this paper, we aim to develop a deep learning based semantic segmentation solution for identifying the 3 healthy and 1 disease category which will aid the autonomous robot to take the decision in real time for strawberry harvesting and disease monitoring. These four categories are unripe, partially ripe, ripe (full) and overgrown/disease. Now although there have been previous attempts for strawberry fruit segmentation, they have some limitations. For example, the segmentation issue was tackled using old image processing techniques which often require manual parameters tuning for each different set of images. These experiments were carried out in a controlled environment instead of the real field environment where lighting, varying background, and occlusion are the big issues. These techniques are also not robust as they were not aimed for the real time applications in the field. Furthermore, the existing literature [4] on strawberries does not discuss the gray mold disease problem which spreads heavily from the overgrown strawberries, which motivated us to investigate this problem. Since the proposed study is for the real-field scenarios, therefore we consider the following few challenges.

1. *Varying field conditions:* Every field has a different environment setup and light intensity conditions. All objects under consideration are not of the same size. For the data acquisition, images are not collected by the camera having the same specifications and resolution. Besides, cluttered background, varying illumination, low contrast between leaves and fruits are also among a few challenges.

2. *Imbalance dataset:* Deep learning algorithms essentially require large training data and there is never enough training data. Also, data annotation for segmentation takes a lot of time and therefore data is limited. Having imbalance distribution (not equal representation of the different classes) can make the learning process biased towards the other classes having a greater number of samples.

3. *Non-uniform data distribution:* Deep Learning algorithms learn upon the pattern of the distribution of the dataset. Whereas this dataset belongs to two different strawberry farms where images of diseased strawberries are altogether captured from a different form, which adds different distribution to the dataset.

To overcome these challenges, we propose a deep convolutional based encoder-decoder network for reliable and precise classification of strawberry fruits into specified categories. Unlike previous approaches which design attention mechanism for channel or spatial attention [4], [14], [15] only. We also focus on designing an attention mechanism for dynamically changing the receptive field (RF) size of neurons depending upon the size of the object. To be specific we design two modules responsible for learning the dynamic receptive field sizes, channel, and spatial attention, as well as a third module for controlling network's computational complexity. These three modules working together enable the convolutional neural network (CNN) to learn both channel and spatial correlations while dynamically changing the RF of neurons for aggregating better multi-scale context and more robust features. We evaluate the performance of our proposed network quantitatively using benchmark metrics and qualitatively using Grad-CAM [4], [16]. Our main contributions can be described as:

- We propose an adaptive receptive field module (ARFM) for dynamically changing the receptive field (RF) size of neurons which helps in better multi-scale context aggregation.
- We design a bottleneck block (BB) to learn channel and spatial interdependencies allowing the network to extract more robust features.
- To reduce the computational complexity and memory footprint of the network we use dilated residual blocks (DRB).
- A new dataset for strawberry fruit segmentation is constructed, having 4 classes based on the ripeness and health of the fruit (see section II "Dataset Construction" for details).

II. LITERATURE REVIEW

The development of a reliable segmentation and detection system is no doubt a challenging task, especially in the real field environment. The need for a robotic system nowadays is inevitable since the agriculture industry is shifting towards technology-intensive from a labor-intensive marketplace. Many attempts have been made via traditional machine learning algorithms in accurately detecting the objects in the agriculture industry [17]. Nguyen *et al.* [18] used RGB-D camera to detect and locate the apples using a color threshold-based technique. They encoded the appearance of red apples using color and geometric features. Further, a Euclidean distance-based clustering algorithm was developed to segment in the feature space. Zhou *et al.* [19] applied the image processing technique in an apple orchard by taking the difference of various color channels. McCool *et al.* [20]

used the Local Binary Patterns in a new way to detect the sweet peppers. Lin *et al.* [21] presented a technique for 3D fruit detection. They represented data into a point cloud and developed a global descriptor vector to capture the important features. Later, an SVM based classifier is utilized to eliminate false positive.

One important problem in agriculture is disease control in an open environment. Strawberries are also prone to pests attack. Ebrahimi *et al.* [22] detected pests in strawberry flowers using SVM based classifier. Huang *et al.* [23] detected insect-damaged samples in Soybeans which utilized multiple statistical image features. Further support vector data descriptor classified the damaged samples. Chung *et al.* [24] pushed forward this research in detecting disease in rice seedlings which is responsible for the healthy growth of rice crops. They developed a Support Vector Machine based classifier which we can say was very popular for building a classifier for classifying the hand engineered features. They also utilized a genetic algorithm to effectively select the optimal parameters. Wheat is one of the most important staples in the world whose health is a big concern in agriculture. Pantazi *et al.* [25], [26] studied the growth cycle and identified biotic and abiotic stresses in wheat crop. These above approaches rely on hand engineered approaches, where classifiers are carefully designed for the task specific problem.

Bosilj *et al.* [27] proposed an image processing based technique for the classification and segmentation in the onion and sugar beet crop. This approach segments the plant regions locally with fine details which is required for the efficient solution. But this technique is relatively slow and does not generalize if a little bit of illumination changes. Potena *et al.* [28] made a robotic system for automatic weed detection as an application in an unmanned ground vehicle. They utilized convolutional neural networks first as a shallow network and then as a deeper network for binary segmentation. Reina *et al.* [29] studied a novel application of terrain assessment in precision agriculture. They not only estimated the terrain using appearance-based features but also physics based features were extracted. Hernández-Hernández *et al.* [30] utilize color models and spaces to eliminate the dependence on illumination conditions to segment the plant and soil. They applied different probability density models to segment the regions and developed their own software tool for the deployment. This approach is relatively easier since there is a significant contextual difference between soil and plant.

Mohanty *et al.* [31] prepared a large repository of plant diseases and work on the identification of various diseases. They utilized existing deep learning architectures from AlexNet and GoogLeNet and produced state of the art results. Sladojevic *et al.* [32] worked on 13 different types of leaf diseases and recognized the diseased samples from the healthy ones. They designed their own network architecture based on CNNs and it was among the first of its kind in terms of the application. Bargoti and Underwood [33] detected location of the fruits classes namely mangoes, almonds, and apples in the orchards. They utilized existing two stage pipeline of famous Faster RCNN for this purpose. But this approach is too slow to be applied with any autonomous vehicle. Another approach of plant phenology was carried out by Yalcin [34] where they collected data of different classes. They utilized pre-trained model of AlexNet and fine-tuned the weights with their newly collected dataset and transformed the learned features for their specific task. Chen *et al.* [35] proposed a fruit counting algorithm by designing their own convolutional neural network for fruit counting. McCool *et al.* [36] segmented the data of crop and weeds for an agricultural robot by designing a lightweight architecture with fewer parameters. Mortensen *et al.* [37] proposed an application of segmentation of mixed crops and weeds dataset. Images were taken by a camera mounted on tractor which moved through the land. A review paper by Kamilaris and Prenafeta-Boldú [38] encompasses the broad vision of the different deep learning approaches used in the agriculture domain. The paper divides the approaches in different sections and gives a brief overview of datasets and the technical details of the contemporary architectures.

Arsenovic *et al.* [39] constructed a largescale plant disease dataset. They proposed a two-stage plant disease net (PD-Net) which further consists of two sub-networks PD-Net1 and PD-Net2. PD-Net1 uses YOLO [40] algorithm to detect plant leaves and PD-Net 2 classify the leaves into different categories. Under real-field condition their method was able to achieve 91.6% mean average precision (mAP) for detection task. Jiang *et al.* [41] proposed a real-time system for apple plant disease and pest recognition. By incorporating InceptionModule [42] and rainbow concatenation (for better multi-scale feature aggregation) with single-stage object detector (SSD) [43]. They were able to achieve 78.8% mAP, with a detection speed of 23.13 frames per second (FPS). Chen *et al.* [44] addressed limitations of classical SLAM (simultaneous localization and mapping) pipeline by proposing a new 3D global mapping system which integrates SLAM and eye-in-hand stereo vision systems. This way their system was able generate a detailed 3D orchard map which can be used for flexible and large-scale orchard picking systems. Nie *et al.* [45] proposed unique method for detecting strawberry verticillium wilt. Instead of directly classifying the whole plant as having verticillium wilt or not, they first classify and detect young petioles and leaves in the image and then used the detected components to decide whether the whole plant is infected or not. They further improve their accuracy by adding a channel attention mechanism in Faster-RCNN's [12] backbone and was able to achieve mAP of 77.54%. Tian *et al.* [46] made substantial changes in YOLO-v3 [47] architecture to detect anthracnose damage in apple plant. They were able to achieve 95.57% mAP by changing the backbone of YOLO-v3 with DenseNet [48] and optimizing feature extraction layer of YOLO-v3. Chen *et al.* [49] proposed a multi-vision system for performing multi-view 3D perception of orchard banana central stock. They installed multiple cameras at different angles to increase framework's field of view (observable scene/ visible view) for better detection results.

Fully Convolution Network (FCN) [50] introduced the earliest popular pipeline for the semantic segmentation. This work leaves the mark on the successor architecture for semantic segmentation and still is relevant. Before FCN, popular classification models were leading the board which includes the giants like AlexNet, VGGNet, and GoogLeNet. FCN transformed these architectures into predicting at pixel level. It utilized the power of transfer learning of these models and by deleting the last fully connected layer, changed these models into fully convolution pipeline and predicted the full resolution mask of the image. It is indeed a great contribution of its kind which opened a new window in the segmentation task and scene understanding. Later came the ParseNet [51] which captured the global information of the scene instead of the region information. It used global average pooling to encode the global features by reducing the feature maps into vectors. This vector which can be said as context vector is normalized with L2 Norm and later feature maps are concatenated. All in all, ParseNet emphasized on the global information such that occluded regions are also predicted as part of the true object class. In 2015 Ronneberger *et al.* [52] proposed a similar architecture as of FCN, which has a contracting and expanding part, called U-Net. The contracting part called encoder encodes the feature map into rich features and the expanding part recovers the spatial information through up-sampling or deconvolution. One of the main contributions U-Net introduced was the concatenation of the cropped features from the down-sampling path to the up-sampling path to avoid losing the precise spatial information. This pipeline has since been a benchmark in designing even the latest state of the art models. At the last stage, 1 × 1 convolution is used to obtain segmentation output. This pipeline is still very popular since the author used it on a very small dataset of 30 images with proper augmentation techniques. It was specially designed for medical images where it is very difficult to get even a small chunk of data. Later on, the intermediate layers were further exploited by Lin *et al.* [53] and they proposed RefineNet. In Refine-Net skip-connections use multipath refinement via different convolutional modules to obtain final predictions. Global Convolutional Network proposed by Peng *et al.* [54] increased the receptive field (RF) size of neurons by factorizing the large convolutional kernels into smaller ones to obtain global contextual embeddings. Zhao *et al.* [55] and Chen *et al.* [56] proposed PSP-Net and Deeplab respectively. The former used spatial pyramid pooling at various scales and the later used atrous convolutions with different dilation rates to exploit multi-scale information.

Deconvolution [57] is another remarkable contribution to the segmentation pipeline. It transforms the feature maps exactly opposite to the convolution operation from the lower dimension to the higher dimension while keeping the same connectivity pattern as of convolution. In transposed convolution (sometimes also called deconvolution), an original size feature map is padded with zeros and the rest of the kernel operation is the same as the convolution. The authors of the paper analyzed the deconvolution and observed that lower-level feature maps attained through deconvolution preserve the spatial location and higher ones are responsible for the class assigning. Chen *et al.* [58] combined dilated convolution with depth-wise sparable convolution and proposed Deeplab v3+. This way they were able to achieve a huge performance boost while keeping model complexity to a bare minimum. Dual attention networks proposed by Fu *et al.* [15] modeled the semantic interdependencies via two-way attention. They designed two bottleneck modules for their network for improved channel and spatial attention, namely CAM (channel attention module) and PAM (position attention module). But did not focus on adjusting the receptive field size of neurons.

Convolutional neural networks (CNNs) attempt to mimic the behavior of the human brain, neurons in the human visual system do not process the entire semantic scene at the same time. Instead, the neurons attempt to process the scenery in order, focusing on only the most important features of the scene in front of them [59]. In contrast to previous works which either try to aggregate multi-scale context or to model the semantic interdependencies via attention mechanism, we propose a new attention mechanism to improve the representational power of our network. Our proposed attention mechanism can (a) dynamically change its receptive filed size to deal with the objects of different scales and at different resolutions, (b) learn both channel and spatial interdependencies and can adaptively prioritize or suppress features according to their significance.

III. DATASET CONSTRUCTION

We collected 2048 × 2048 resolution images from two different farms with a camera mounted very close to the field of view. This data is first of its kind, for firstly there is no publicly available dataset to work on this problem, secondly, the already published works on strawberries do not take account of diseased strawberries. Since our dataset is from different farms, there is a variety of background, fruit size, and illumination (sunny, cloudy, etc.). We believe it will further help in developing autonomous agriculture applications. Images are taken in the full blossom harvesting season i.e., between March 2019 to June 2019, in the suburbs of the Jeollabuk-do province of South-Korea. We collected more than 700 images and randomly filtered out 410 images for the experiments. We split the dataset into 3 parts: training, validation, and testing images. We used 281 images for training, 55 for validation, and 74 for testing purposes. Table 1 lists the number of instances present in dataset belonging to each class. Furthermore, we deploy various data augmentation techniques to increase the number of samples in the dataset (see section V "Experimental Setup" for details). All the comparative experiments on the other

TABLE 1. Strawberry segmentation dataset.

Classes	Train	Test	Valid	Total no. of Instances
Ripe	405	163	87	655
Partially Ripe	193	50	21	264
Unrip	686	220	101	1007
Diseased/Overgrown	65	52	20	137
Number of Images	281	74	55	-

state of the art algorithms are conducted on the same pattern of images. There are 4 categories in the dataset which are unripe, partially-ripe, ripe, and overgrown or diseased. Each image was later on resized to 512 × 512 pixels resolution. A subsample of the dataset is shown in Figure 1 (a). We can observe the cluttered background, multiple fruit overlapping and high contrast in the data. Figure 1 (a) shows the data acquired from the different farms varies heavily and colors of the same class are not consistent. Moreover, the samples of the diseased strawberries are difficult to obtain, therefore, we assigned the same class to the overgrown and diseased category. The diseased or overgrown class data is primarily collected from another farm which is responsible for adding an entirely new distribution to the existing dataset. We know that uniform distribution is highly required for the learning of convolutional neural networks and such type of data will make it hard for the weights/learning parameters to adapt to the representation. Therefore, this change in distribution makes the task challenging. Lastly, the color pallet used for representing different strawberry categories is shown in Figure 1 (b).

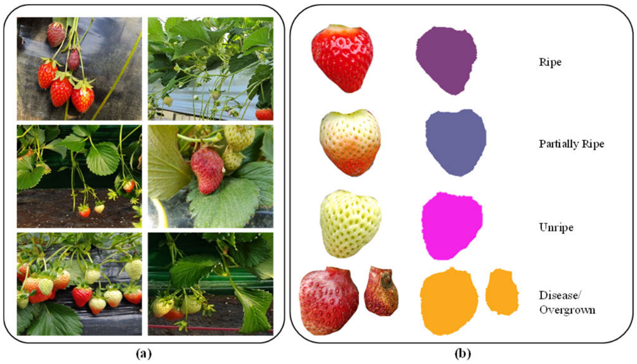


FIGURE 1. (a) Subsample of dataset, showing different background, lighting conditions, multiple fruit overlapping etc. (b) Color pallet used for labelling pixels of a specific class.

IV. PROPOSED ARCHITECTURE

One of the core purposes of the computer vision approaches is to build powerful representations that extract only those salient features and properties from an image that are suited for the given task and hence improving performance. We have utilized various properties of the feature representing modules as per our task's need. The flow of the complete network is like a typical encoder and decoder, but each stage has independent representation during the flow of information. Let us take the standard U-Net architecture to understand our proposed architecture. U-Net down-samples the input image, with 2 convolutions at each stage, and keeps on increasing the number of channels until just before the up-sampling stage. In the decoder part, the number of stages is the same as the encoder part. Connections from each corresponding stage across the encoder and decoder are made to concatenate the feature maps. But this structure was specially designed for only medical images, which is used to extract dense information and is transmitted across the network. Now in our architecture, there is a number of design changes we have made according to our problem. Since our input size is 512 × 512 and it does contain contextual RGB information, so we restrict ourselves to pool (subsample) the image 4 times only thereby reducing its size to 64 × 64 after successive down-sampling stages.

feature map in their spatial dimension.

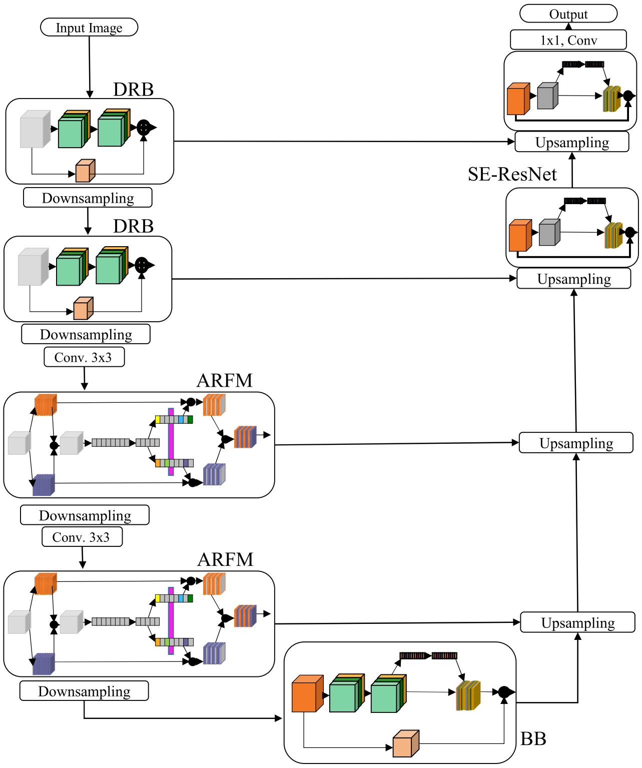


FIGURE 2. Complete architecture of proposed convolutional encoder-decoder network.

Detailed and complete architecture is represented in Figure 2. For the first two stages of the encoder part, we used depth-wise residual blocks instead of the regular convolution. For the subsequent two stages in the encoder, we used an "adaptive receptive field module" with two back-to-back convolution operations before this module. Now later at the end of the encoder stage, we used the bottleneck module which is followed by the decoder part. We did not perform any further pooling operations in the bottleneck where the core purpose was to capture the rich intrinsic representations before entering the decoder part. We kept the decoder design very plain except for the last two stages. In this part, the number of stages is the same as the encoder. In each stage, a simple bilinear operation is performed to up-sample the feature map in their spatial dimension.

At each stage, the spatial dimension matches the dimension of the corresponding stage in the encoder part, and feature maps of each stage are concatenated. This concatenation is very important as the feature maps carry the spatial location of the objects. For the last two stages, we used SE-ResNet blocks [14], [60]. The rationale behind is that after concatenation of the features from the encoder stages and the up-sampled feature maps from the previous decoder stages we have to choose those channels which are not correlated and do not carry redundant information. SE-ResNet blocks also provide attention to the channels in a class-specific manner. As the last step, 1×1 convolution is applied to match the number of class categories as an output. Detail analysis, usage, rationale, and structure are presented individually in the following subsections. If we talk about the network complexity, then there are parameters comparative to the existing state of the art.

A. DEPTH-WISE RESIDUAL BLOCK (DRB)

For the first two stages in the encoder (left half) in Figure 2, we used depth-wise residual blocks instead of regular convolution. In regular convolution 1 filter having the depth of the input tensor is convolved on the input tensor. So, to get the required number of channels in the output, we have to use the same number of filters each having the same depth as the input tensor. Whereas, in depth-wise convolution, each filter having the depth of 1 is convolved with its corresponding feature map only and the resultant feature map is spatially enhanced. Now, these spatially enhanced feature maps of distinct filters are stacked to get the final output. This is very cheap in terms of computation than the regular convolution. The number of distinct filters is the same as the number of feature maps in the input tensor.

In our architecture, we use depth-wise residual blocks with 2 depth-wise convolution operations in each block in the encoder part. As compared to the standard U-net stage in the encoder part, we used this block to get the distinct feature map with less floating point (addition and subtraction) operation. It computes two depth-wise convolutions back-to-back in each block with kernel size 3. Later we add the resultant feature map with the input feature map to compute the final output. There are two stages in the encoder part where depth-wise Residual block is used. Both stages end with a pooling operation.

B. ADAPTIVE RECEPTIVE FIELD MODULE (ARFM)

Neuroscience has greatly inspired the design of convolutional neural networks. In neuroscience, the receptive field size depends on the stimulus received by neurons [61], [62]. So, the neurons adapt to the receptive fields itself through the stimulus [59]. This has been unexplored in the CNN architectures. Many of the approaches starting from the inception module by GoogLeNet and later versions have considered the variable size receptive field for aggregating distinct features captured by the customized kernel size. Similarly, ResNext introduces cardinality for group convolution which characterizes both depth and width. Additionally, Xception network introduces depth-wise separable convolution which is composed of depth-wise convolution and pointwise convolution. This proves its effectiveness in reducing parameters yet with extracting good features. Likewise transpose convolution which raise the receptive field exponentially by keeping parameters constant. Additionally, although we can get a larger receptive field but this sometimes comes with gridding effect. Therefore, instead of using custom designed kernels with their linear aggregation of feature maps, we utilize "adaptive receptive field module" in our network. The module was originally introduced by "Selective Kernel Networks" [63] for classification purpose, but we introduce this in the semantic segmentation pipeline and determine its best position for adopting it in the encoder decoder architecture. As described earlier, the main purpose of the module is to learn to adapt to the receptive field based on the stimulus received.

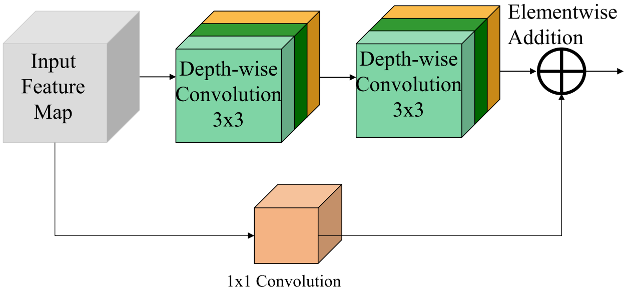


FIGURE 3. Depth-wise residual block for initial layers of encoder as a replacement to the dense regular convolution.

This module receives deep tensor as an input and outputs a feature map which contains more distinct features selected by an adaptive mechanism. The module performs three basic operations: Split, fuse and select. First part of the module splits features maps by different kernels. These number of kernels can vary but we choose two here. In the next step, these computed feature maps are linearly added, and global average pooling is performed. This global average pooling transforms the feature maps into fully connected neurons. This first fully connected layer further connects to another fully connected dense layer which has number of neurons as the fraction of the first layer numbers.

C. BOTTLENECK BLOCK (BB)

In semantic Segmentation pipeline, bottleneck block is very important stage since it controls the flow of information from continuous down-sampling part to the up-sampling part. Contrary to the ASPP block of the DeepLab architecture [56], we have used the bottleneck block. The purpose of this block

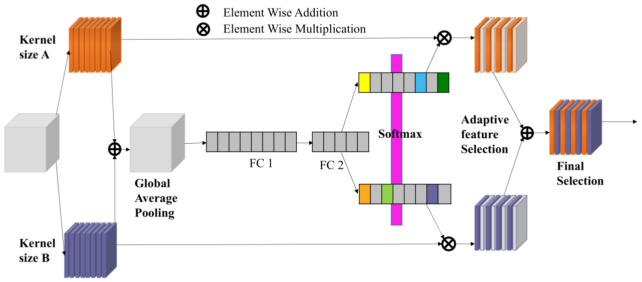


FIGURE 4. Adaptive feature selection module to automatically select the suitable receptive field.

is to pass the rich extracted features from first part of the network to the second. Successive down-sampling produces feature maps of reduced spatial dimension with large number of channels. This block computes the distinct features from the encoder part and does not down-sample any further as we restrict ourselves to 4 times down-sample w.r.t the input size. It propagates those features which has less correlation with each other.

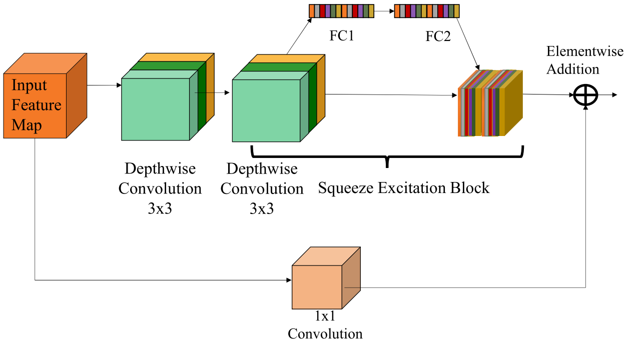


FIGURE 5. Bottleneck block used at the junction of encoder and decoder part.

The block consists of two back-to-back depth-wise convolutions (for enhancing feature maps spatially) followed by a squeeze excitation sub block (for remodeling channel interdependencies). So, this overall block is the modified form of the SE blocks. A skip connection adds the input tensor elementwise with the aggregated output of the squeeze excitation sub block. Depth-wise convolutions computes features which rely only on the respective filters and do not attribute to the features present in the depth. These responses are enhanced by the squeeze and excitation module which provides adaptive recalibration to the channels. This module mitigates the response maps which are highly correlated, and the module also works as an attention module to emphasize the strong individual features.

D. SQUEEZE EXCITATION RESNET BLOCKS

In the decoder part, we have used the Squeeze Excitation ResNet (SE-ResNet) blocks originally used in the paper for classification tasks [14]. We reuse this module for the segmentation purpose. We use a specific property of SE-ResNets in our task. The author of SE-Net claims that in the later layers of the network SE blocks very much work in class specific manner such that they adaptively learn to respond to the inputs. Attention mechanism is another essential property of the SE-ResNet blocks which guides our choice to use these blocks in the later layers of the decoder part. This block is inevitable in our decoder choice. In the decoder part, the feature maps output of the bottleneck block is up sampled directly two times to match the dimension from its counterpart in the encoder part which are then concatenated. Afterwards for the last two stages, we up sampled and concatenated the feature maps, same as the previous two stages but here we applied SE-ResNet Blocks. Since, until this stage, many feature maps are redundant and carry similar information across the depth, therefore this block adaptively selected those channels which are not correlated. It also gives attention to the feature map with respect to each class category.

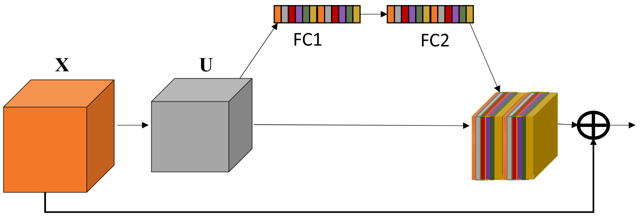


FIGURE 6. Squeeze and excitation ResNet (SE-ResNet) block to select the important channels and to give attention.

V. EXPERIMENTAL SETUP

The experimental setup is descried in this section. Then we describe the evaluation metrics used to evaluate the performance of proposed framework.

A. IMPLEMENTATION DETAILS

All the experiments reported are carried out with the following data splitting of total 410 images. We used 281 images for training, 55 for validation and 74 for test results. The experiments are conducted having CPU specification as Intel Core i9-9940X, 3.3GHz Processor and 128 GB RAM. GPU utilized is NVIDIA RTX 2080. Since the data is not in a huge amount, therefore we have trained for 14000 iterations. To avoid memory exceeding, we used batch size of 2. Adam is used to optimize the parameters with learning rate 0.0001 and momentum value of 0.9. We used two different learning rate schedules to decrease the learning rate after each epoch: time decay and step decay. Learning rate schedules are used along with momentum to keep the learning process smooth. This learning rate changes over the iterations and instantiate the networks with a new learning rate. In this way although the network training is not confined to a single learning rate, but still initial learning rate matters.

To reduce training time and computational requirements, we resized all the images and segmentation masks to 512 × 512 resolution without preserving the aspect ratio before training. During training, we also did substantial on-the-fly data augmentation (augmenting data at each iteration) to increase the dataset size and avoid overfitting. In terms of augmentation techniques, we only used those that were appropriate for segmentation problems and improved the network's robustness. Random cropping and resizing, random mirroring along the vertical axis, random rotation, and finally, random brightness and saturation distortion were all used.

Furthermore, we used early stopping with a condition to stop the training once a certain condition is met. We used two different conditions of early stopping: one with the validation intersection over union and one with validation loss. For example, if our validation loss does not decrease over the course of few epochs, the condition stops the training, and we can analyze which hyper parameter we need to tune in order to get the training going on.

B. EVALUATION METRICS

1) JACCARD INDEX

For semantic segmentation, Jaccard Index also referred as intersection over union is the most widely used metric to evaluate the performance of the network on the dataset. Semantic segmentation, unlike classification is a dense pixel-wise prediction, so each of the pixel label contributes in evaluating. It calculates the degree of overlap between the ground truth images and predicted masks.

$$IoU = \frac{target \cap prediction}{target \cup prediction} \tag{1}$$

In equation 1, numerator term calculates the common pixels (intersection) in both target and prediction whereas the denominator term calculates union of both target and prediction. In semantic segmentation, it is common practice to calculate IoU in binary class form. For example, we have 4 object classes and 1 background class. Now our targets and ground truth both are in 5 different channels. Each channel in prediction will be compared to its corresponding ground truth channel to compute individual IoU and then mean over all the classes will give the resultant value. This metric gives equal weightage to all the channels.

2) PRECISION, RECALL AND F1 SCORE

Beside the intersection over union, we used Precision, Recall and F1 Score to evaluate our network performance. Precision and recall both somewhat explain the evaluation related to the accuracy but not exactly the same. Precision evaluates that the how much part of the prediction is relevant while recall tells us how much percentage of the total relevant results are correctly classified by the network. Now, network has to find the tradeoff between precision and recall as per the given task whether we need to maximize precision or recall. F1 score calculates kind of harmonic mean of the precision and recall. We have calculated precision and recall individually for the 4 class categories whereas to calculate the F1 Score, we computed the mean of the 4 classes.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive} \tag{2}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative} \tag{3}$$

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

TABLE 2. Quantitative comparison of results with other state of the art segmentation networks.

Network Architecture	mIoU (%)	F1-Score (%)	Param. (Million)	FPS (sec)
U-Net	74.52	0.7818	30.9	23.30
SegNet	53.53	0.5137	29.44	9.21
DeepLabv3	78.72	0.7628	3.5	32.67
DeepLabV3+	76.52	0.7961	3.8	**35.21**
DAN	59.01	0.5732	**1.7**	15.27
Proposed	**81.94**	**0.8292**	4.6	28.37

VI. RESULTS AND DISCUSSION

In the following Table 2, the network performance can be seen in terms of the Intersection over Union and F1 score. Number of parameters are also mentioned against each network. It can be observed in Table 2 that the proposed architecture outperforms the current state of the art deeplabv3+ [64] and dual attention network (DAN) [15] in both mean Intersection over Union and F1 score. However, there is a slight increase in the parameters with deeplab variants. This increase is not even close to the U-Net and SegNet approaches. With rigorous experimentation of tuning hyper-parameters in the U-Net and SegNet, the performance could not increase. We report that this difference is because of the significant change of data distribution as the datasets belong to different farms. As we have mentioned in the preceding section while introducing the dataset that we have gathered the dataset from two different places with entirely different camera sensors and changed lightning conditions. Even the season of the crop was different. 3 classes of healthy strawberries unripe, ripe and partially ripe belong to the different dataset while overgrown/disease class has a different distribution. All these factors contributed to reduce the performance of the previous state of the art models and hence pave the way for us to build a new network that can well handle the changing crop conditions that are real and varying. By looking at the evaluation metric table, we can claim that our network performs better than the existing approaches.

The latency is another important property when it comes to the deployment. This is essential since decision must be

TABLE 3. Precision and recall values for individual classes at confidence threshold of 0.5. The bold is used to represent best results.

Network Architecture	Precision at Confidence Threshold 0.5				Recall at Confidence Threshold 0.5			
	Partially Ripe	Ripe	Unripe	Overgrown/Disease	Partially Ripe	Ripe	Unripe	Overgrown/Disease
U-Net	0.7259	0.9228	0.8663	0.9506	**0.5839**	0.8982	0.7477	0.6191
SegNet	0.4456	0.7288	0.7383	0. 544	0.5064	0.799	0.7051	0.43
DeepLabv3	**0.9357**	0.8899	0.8564	**0.9664**	0.4388	0.9593	0.8423	0.3818
DeepLabV3+	0.7687	0.9179	0.7734	0.9527	0.5036	0.9366	**0.9366**	0.7435
DAN	0.3964	0.8727	0.4507	0.8549	0.3811	0.8385	0.8462	0.07
Proposed	0.7885	**0.9232**	**0.8873**	0.9245	0.5734	0.9282	0.8786	**0.7535**

taken in the real time. In machine vision tasks, deep learning-based network will identify the regions of the strawberry class which will further be harvested by the robotic part. In Table we report that our network processes 28.37 frames per second.

Following Figure 7 shows the precision and recall of individual 4 classes. Two classes ripe and unripe show excellent behavior but the two classes partially ripe and disease fall a little short behind. We can explain this phenomenon as the number of instances in partially ripe and disease classes are less than the other two (refer to Table 1 for details). Moreover, apparently, partially ripe class on one hand resembles with the unripe class slightly whereas it has the features of the ripe class on the other hand. This phenomenon also is responsible for the confusion between the classes.

In the same way, we also report the confusion matrix in Figure 8 which further analyzes the network response to the individual classes. It is one of the explicit ways to know the contribution of the individual class in overall decision making of the network. As the name suggests, confusion matrix tells us how much network is confused and wrongly predicts the one class for the other. Diagonal entries correspond to the True Positive values for each class. For example, 3rd diagonal entry has the true label "ripe" on the y-axis and predicted label "ripe" in the x-axis. Now the value 0.93 tells us that 93 percent pixels of ripe class were truly predicted as "ripe", and in the same row for 1% pixels, network confused "ripe" with "diseased" class. In the same way, for the same "ripe" class, network wrongly said that 4% pixels belong to the "partially ripe" and 2% were wrongly predicted as background. In the Figure 8, we can see that, ripe class produces best results. This class has the least confusion of its pixel percentage with the other classes. Whereas there is problem with partially ripe and disease class. In the latter case, number of class instances (overall pixel representation in the data) are relatively less as compared to the other classes. Moreover, since all the images containing the diseased class belong to a different dataset, therefore having non-uniformity in the dataset caused the confusion in the prediction.

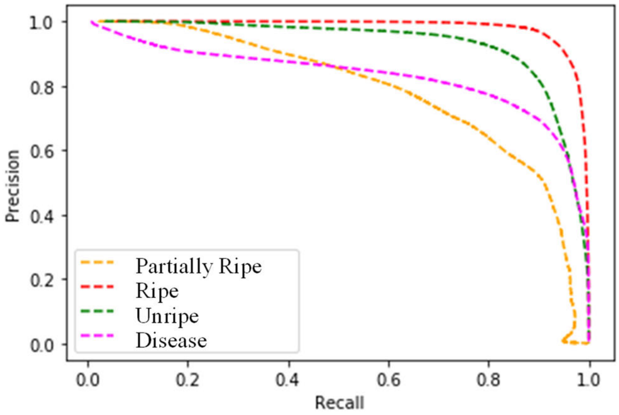
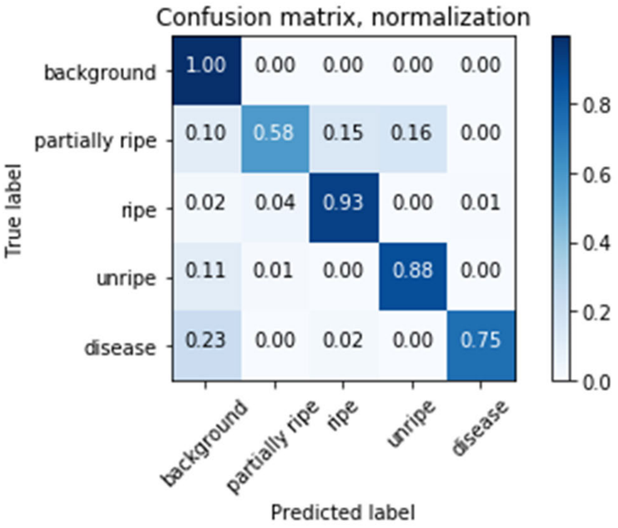


FIGURE 7. Precision and recall curve for individual classes.



	background	partially ripe	ripe	unripe	disease
background	1.00	0.00	0.00	0.00	0.00
partially ripe	0.10	0.58	0.15	0.16	0.00
ripe	0.02	0.04	0.93	0.00	0.01
unripe	0.11	0.01	0.00	0.88	0.00
disease	0.23	0.00	0.02	0.00	0.75

FIGURE 8. Confusion matrix of proposed network for individual classes.

Another quantitative analysis of the network performance is reported in the Table 3 where we statistically show the precision and recall values for the 4 classes individually.

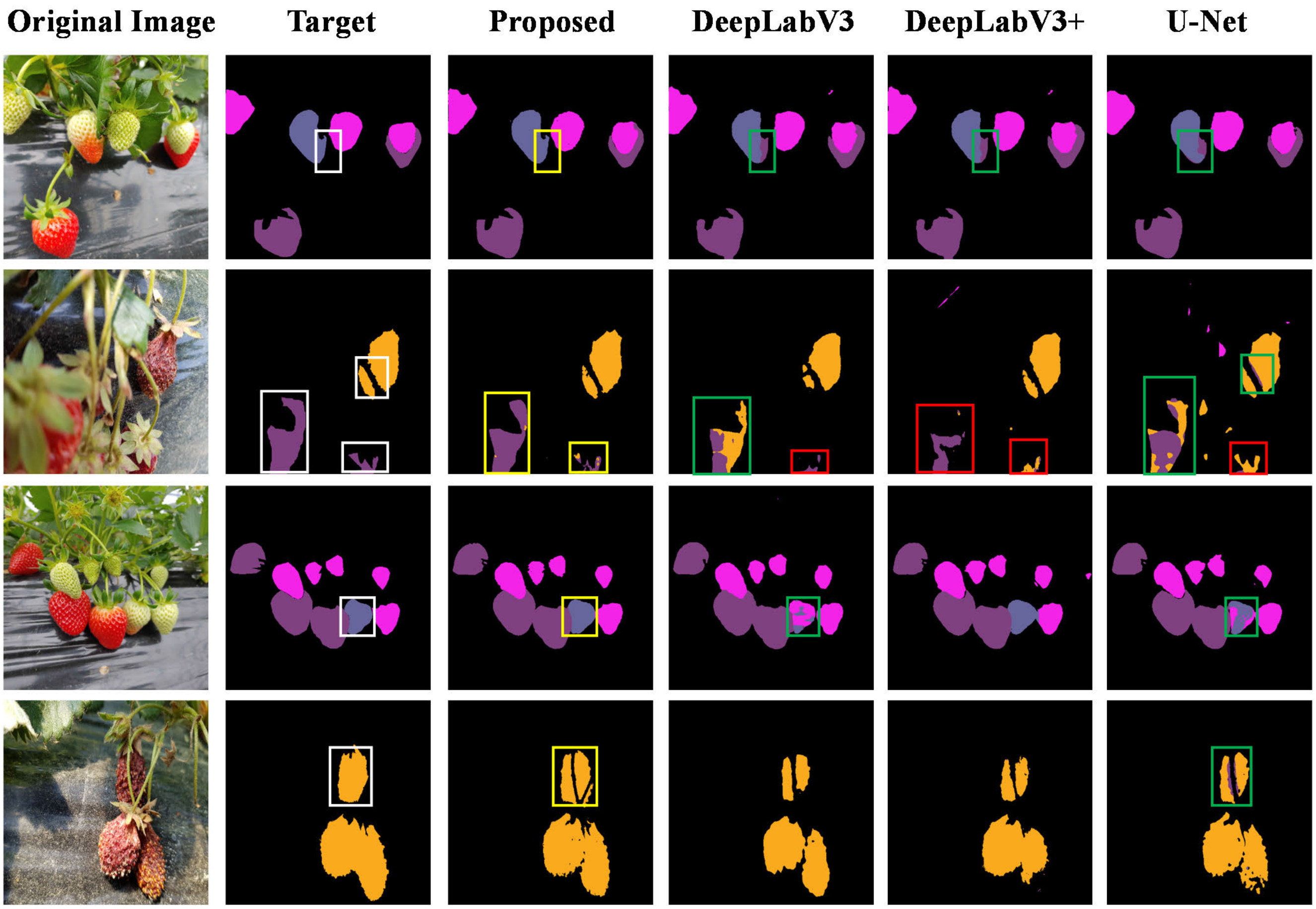


FIGURE 9. Qualitative comparison of results on strawberry segmentation dataset. Whit boxes show the areas of interest (for sake of discussion) in the ground truth labels. Yellow boxes highlight the areas where proposed network clearly outperforms the previous segmentation works. Green boxes show one fruit being classified as belonging to two different classes. Red boxes represent partially misclassified instances and false negative instances.

We can see that cumulative effect of our network is dominant in both recall and precision values that shows the effectiveness of our network. DeepLabv3 shows good precision values in partially ripe and diseased classes, but simultaneously the recall values for the same two classes are very poor. It is highly likely that for higher precision value, recall value at the same time could be very value. It depends upon the application whether we need high recall or high precision. In this work we try to find a balance between precision and recall values. As we discussed before about the varying data distribution particularly in the diseased class cause the other networks to perform poorly on the recall value of the diseased category. Therefore, the value for the dual attention network is very small.

A. QUALITATIVE RESULTS

We evaluate the performance of our network qualitatively for the semantic segmentation of the strawberry classes. In the Figure 9, we show a broad qualitative comparison with other networks. First column represents the original images whereas target is the ground truth label which the models try to achieve. Column three to six show the outputs of their respective networks as shown in Figure 9. White boxes in 1st column of Figure 9 shows some areas of interest that clearly demonstrate our network's superior performance when compared to others. Green boxes in columns 4,5 and 6 show that a single strawberry fruit is being detected as belonging to two different classes. Whereas red boxes in columns four to six show either partial detection or misclassification of strawberry fruit instance. In contrast the yellow boxes in column three show that our proposed attention mechanism successfully captures the right scene context and recognizes the objects correctly. Row two of Figure 9 supports our argument where one fruit instance is very close to the camera and the other is far away. Even so, due to the adaptive receptive field of our ARFM module the fruit instance is correctly classified as comparted to other networks.

Furthermore, labelling noise (assigning wrong labels while annotating the data) is also present in the dataset, to avoid overfitting. An example of labelling noise is shown in last row of Figure 9. Here, two runners occlude the diseased category strawberry, but the target label include pixels belonging to the runners (that should be in background) in the diseased category. Here also proposed network sharply detected the boundaries across the runners by only predicting the pixels that belong to the diseased category. Other networks also try to avoid the labelling noise, but they are not very good at it.

B. FEATURE VISUALIZATION

The role of convolution neural networks in image recognition tasks cannot be undermined. Designing of the networks have become very mature field since the past few years. However, an interpretation of these network becomes inevitable that how a certain design is serving our specific task's needs and how the neurons in the deeper layers of CNN react to a particular input. Following [4] we use segmentation Grad-CAM (SGC) to visualize the activation heatmaps of neurons in the last layer of proposed network (i.e., after last SE-ResNet block). Grad-CAM tries to intercept the rationale behind the decision taken by the convolutional neural network. Figure 11 shows how the neurons of the last convolution layer of our proposed CNN react to each of the class present in the dataset. Our proposed CNN has four output channels one for each class in the dataset. Given the inputs shown in column 'a' of Figure 11, activation maps of each channel corresponding to a specific class are shown in column 'c' of Figure 11. The column 'b' of Figure 11 shows the channel activation maps of our baseline model shown in Figure 10. It can be clearly seen form Figure 11 (last two rows) that our enhanced attention mechanism refines the activations of the neurons in each channel allowing the neuron of each layer to focus on their respective targeted class.

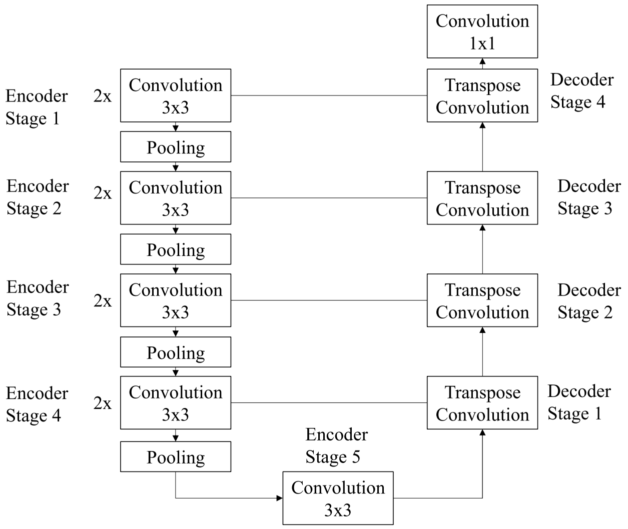


FIGURE 10. Baseline model for ablation study. This model has four convolution stages for consecutive down-sampling and 5th stage as a bottleneck. Changes have been made by replacing modules step by step.

TABLE 4. Ablation experiments by replacing the modules in the baseline model.

Network	DRB	ARFM	BB	SE-ResNet	mIoU	Param. (M)
Baseline					0.7432	6.52
			✓		0.7646	5.56
	✓				0.7457	3.66
		✓			0.7816	6.52
				✓	0.7321	3.72
		✓		✓	0.7852	6.53
	✓	✓			0.7712	6.47
	✓		✓		0.7059	1.76
	✓	✓		✓	0.7632	4.58
Proposed	✓	✓	✓	✓	**0.8194**	4.63

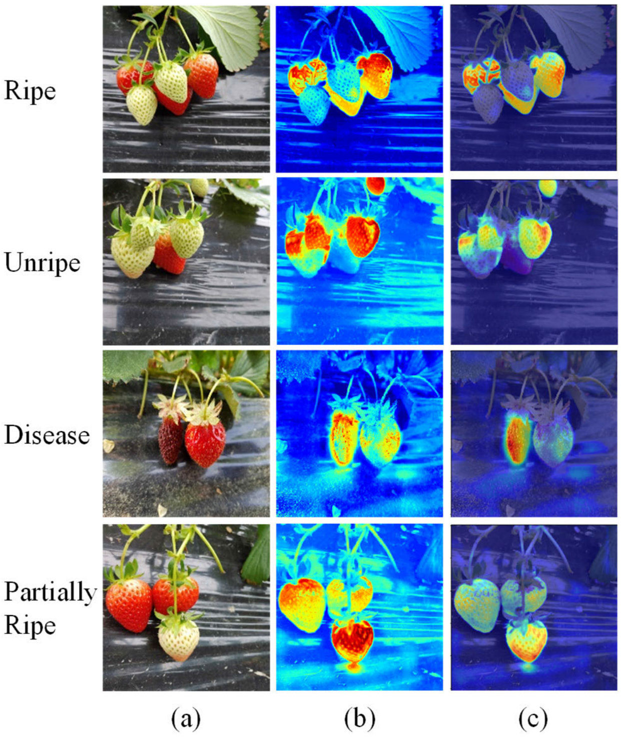


FIGURE 11. Visualization of heatmaps, generated via segmentation Grad-CAM for displaying effectiveness of attention mechanism. In heatmaps 'red' color means highest activation and 'blue' color means no activation at all. (a) image input to network, (b) heatmaps generated by baseline network and (c) heatmaps generated by proposed network.

C. ABLATION EXPERIMENTS

The proposed architecture consists of different modules as shown in Figure 2. We evaluate the performance of each module in Table 4. Baseline module is the encoder decoder U-Net like architecture (Figure 10) which has four stages of convolution blocks for down-sampling and again a convolution block at the bottleneck 5th stage just before the decoder part. In ablation experiments, we have replaced the simple convolution block stages in the baseline model by gradually replacing our modules. For example, in first experiment we replace the 1st two convolution blocks with DRB, in second experiment we replace the next two convolution blocks with ARFM module and so on. Adaptive Receptive Field Module (ARFM) is used two times in the proposed network. Bottleneck block (BB) is used only once and squeeze-excitation block (SE-ResNet) is used two times in the last two decoder stages whereas the first two decoder stages are replaced with simple bilinear upsampling layers. In the Table 4, we report the impact of replacing each module one by one, as well as the effect of combining two or more modules to draw a comprehensive picture. Replacing convolution layer at the bottleneck alone improves the mean IoU by 2%. Replacing the 1st tow encoder stages with DRB results in decreased parameters but performs stays roughly same.

When used in conjunction with other modules, the DRB module performs really well. Like SE-ResNet module, it performs best when used in conjunction with other modules. Even if all the modules are placed except bottleneck, then also performance drops significantly. Finally, all the modules work together to produce better results with controlled parameters.

VII. CONCLUSION

In this paper we have proposed a novel and efficient deep learning approach for the semantic segmentation of healthy and disease/overgrown strawberries for harvesting purpose in machine vision tasks. The system introduces adaptive receptive field and channel selection modules which give the network ability to tackle the variable sized instances and correlated feature maps. The bottleneck module computes the rich feature while transforming the information from encoder part to decoder part. We present a dataset with a high degree of non-uniformity in the distribution, as images are from various environments with varying camera sensors, illumination, and focal length. The visualization of intermediate layers in the network shows the effectiveness of the modules used. We evaluate our network on the proposed dataset. The network overall achieves 3% increased performance in Intersection over Union as compared to the modern state of the art models yet maintaining the real time performance. It produced great results in coping up with the highly occluded case scenarios. In addition, proposed network has the capability of deployment in the real crop condition due to its performance in varying and dynamic environment. The system will help prevent the spreading of gray mold disease by timely identifying and removing the infected strawberries. As the diseased class data of strawberries is hard to achieve, so we aim to collect this in our future work to include more class categories of the different diseases, hence making dataset more dynamic. Due to the nature of semantic segmentation task, currently the proposed network might be unable to isolate individual instances. But our future work involves integration of proposed method with depth-estimation and semantic graphics for improve localization and recognition of individual fruit instances. Finally, we believe that this work will help taking smart farming a step further and will also be of great help for future researchers in harvesting, fighting disease and surveillance.

REFERENCES

[1] W. R. Jarvis, "The infection of strawberry and raspberry fruits by *Botrytis cinerea* Fr," *Ann. Appl. Biol.*, vol. 50, no. 3, pp. 569–575, Sep. 1962.

[2] M. Leroch, C. Plesken, R. W. S. Weber, F. Kauff, G. Scalliet, and M. Hahn, "Gray mold populations in German strawberry fields are resistant to multiple fungicides and dominated by a novel clade closely related to *Botrytis cinerea*," *Appl. Environ. Microbiol.*, vol. 79, no. 1, pp. 159–167, Jan. 2013.

[3] N. U. Islam and J. Park, "Depth estimation from a single RGB image using fine-tuned generative adversarial network," *IEEE Access*, vol. 9, pp. 32781–32794, 2021.

[4] T. Ilyas, M. Umraiz, A. Khan, and H. Kim, "DAM: Hierarchical adaptive feature selection using convolution encoder decoder network for strawberry segmentation," *Frontiers Plant Sci.*, vol. 12, p. 189, Feb. 2021.

[5] M. U. Rehman, S. Cho, J. H. Kim, and K. T. Chong, "BU-Net: Brain tumor segmentation using modified U-Net architecture," *Electronics*, vol. 9, no. 12, p. 2203, Dec. 2020.

[6] S. D. Ali, W. Alam, H. Tayara, and K. Chong, "Identification of functional piRNAs using a convolutional neural network," *IEEE/ACM Trans. Comput. Biol. Bioinf.*, early access, Oct. 29, 2020, doi: 10.1109/TCBB.2020.3034313.

[7] A. Khan, T. Ilyas, M. Umraiz, Z. I. Mannan, and H. Kim, "CED-Net: Crops and weeds segmentation for smart farming using a small cascaded encoder-decoder architecture," *Electronics*, vol. 9, no. 10, p. 1602, Oct. 2020.

[8] M. U. Rehman, Z. Abbas, S. H. Khan, S. H. Ghani, and Najam, "Diabetic retinopathy fundus image classification using discrete wavelet transform," in *Proc. 2nd Int. Conf. Eng. Innov. (ICEI)*, Jul. 2018, pp. 75–80.

[9] M. Shujaat, A. Wahab, H. Tayara, and K. T. Chong, "PcPromoter-CNN: A CNN-based prediction and classification of promoters," *Genes*, vol. 11, no. 12, p. 1529, Dec. 2020.

[10] M. Shujaat, S. B. Lee, H. Tayara, and K. T. Chong, "Cr-prom: A convolutional neural network-based model for the prediction of rice promoters," *IEEE Access*, vol. 9, pp. 81485–81491, 2021.

[11] W. Alam, S. D. Ali, H. Tayara, and K. Chong, "A CNN-based RNA N6-methyladenosine site predictor for multiple species using heterogeneous features representation," *IEEE Access*, vol. 8, pp. 138203–138209, 2020.

[12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 91–99.

[13] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2961–2969.

[14] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 7132–7141.

[15] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3146–3154.

[16] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 618–626.

[17] Y. Tang, M. Chen, C. Wang, L. Luo, J. Li, G. Lian, and X. Zou, "Recognition and localization methods for vision-based fruit picking robots: A review," *Frontiers Plant Sci.*, vol. 11, p. 510, May 2020.

[18] T. T. Nguyen, K. Vandevoorde, N. Wouters, E. Kayacan, J. G. De Baerdemaeker, and W. Saeys, "Detection of red and bicoloured apples on tree with an RGB-D camera," *Biosyst. Eng.*, vol. 146, pp. 33–44, Jun. 2016.

[19] R. Zhou, L. Damerow, Y. Sun, and M. M. Blanke, "Using colour features of cv. 'Gala' apple fruits in an orchard in image processing to predict yield," *Precis. Agricult.*, vol. 13, no. 5, pp. 568–580, Oct. 2012.

[20] C. McCool, I. Sa, F. Dayoub, C. Lehnert, T. Perez, and B. Upcroft, "Visual detection of occluded crop: For automated harvesting," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2016, pp. 2506–2512.

[21] G. Lin, Y. Tang, X. Zou, J. Xiong, and Y. Fang, "Color-, depth-, and shape-based 3D fruit detection," *Precis. Agricult.*, vol. 21, no. 1, pp. 1–17, Feb. 2020.

[22] M. Ebrahimi, M. Khoshtaghaza, S. Minaei, and B. Jamshidi, "Vision-based pest detection based on SVM classification method," *Comput. Electron. Agricult.*, vol. 137, pp. 52–58, May 2017.

[23] M. Huang, X. Wan, M. Zhang, and Q. Zhu, "Detection of insect-damaged vegetable soybeans using hyperspectral transmittance image," *J. Food Eng.*, vol. 116, no. 1, pp. 45–49, May 2013.

[24] C.-L. Chung, K.-J. Huang, S.-Y. Chen, M.-H. Lai, Y.-C. Chen, and Y.-F. Kuo, "Detecting Bakanae disease in rice seedlings by machine vision," *Comput. Electron. Agricult.*, vol. 121, pp. 404–411, Feb. 2016.

[25] X. E. Pantazi, D. Moshou, R. Oberti, J. West, A. M. Mouazen, and D. Bochtis, "Detection of biotic and abiotic stresses in crops by using hierarchical self organizing classifiers," *Precis. Agricult.*, vol. 18, no. 3, pp. 383–393, Jun. 2017.

[26] X. E. Pantazi, A. A. Tamouridou, T. K. Alexandridis, A. L. Lagopodi, J. Kashefi, and D. Moshou, "Evaluation of hierarchical self-organising maps for weed mapping using UAS multispectral imagery," *Comput. Electron. Agricult.*, vol. 139, pp. 224–230, Jun. 2017.

[27] P. Bosilj, T. Duckett, and G. Cielniak, "Connected attribute morphology for unified vegetation segmentation and classification in precision agriculture," *Comput. Ind.*, vol. 98, pp. 226–240, Jun. 2018.

[28] C. Potena, D. Nardi, and A. Pretto, "Fast and accurate crop and weed identification with summarized train sets for precision agriculture," in *Proc. Int. Conf. Intell. Auton. Syst.* Cham, Switzerland: Springer, 2016, pp. 105–121.

[29] G. Reina, A. Milella, and R. Galati, "Terrain assessment for precision agriculture using vehicle dynamic modelling," *Biosyst. Eng.*, vol. 162, pp. 124–139, Oct. 2017.

[30] J. L. Hernández-Hernández, G. García-Mateos, J. M. González-Esquiva, D. Escarabajal-Henarejos, A. Ruiz-Canales, and J. M. Molina-Martínez, "Optimal color space selection method for plant/soil segmentation in agriculture," *Comput. Electron. Agricult.*, vol. 122, pp. 124–132, Mar. 2016.

[31] S. P. Mohanty, D. P. Hughes, and M. Salathé, "Using deep learning for image-based plant disease detection," *Frontiers Plant Sci.*, vol. 7, p. 1419, Sep. 2016.

[32] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," *Comput. Intell. Neurosci.*, vol. 2016, pp. 1–11, May 2016.

[33] S. Bargoti and J. Underwood, "Deep fruit detection in orchards," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, May 2017, pp. 3626–3633.

[34] H. Yalcin, "Plant phenology recognition using deep learning: Deep-pheno," in *Proc. 6th Int. Conf. Agro-Geoinform.*, Aug. 2017, pp. 1–5.

[35] S. W. Chen, S. S. Shivakumar, S. Dcunha, J. Das, E. Okon, C. Qu, C. J. Taylor, and V. Kumar, "Counting apples and oranges with deep learning: A data-driven approach," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 781–788, Apr. 2017.

[36] C. McCool, T. Perez, and B. Upcroft, "Mixtures of lightweight deep convolutional neural networks: Applied to agricultural robotics," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1344–1351, Jul. 2017.

[37] A. K. Mortensen, M. Dyrmann, H. Karstoft, R. N. Jørgensen, and R. Gislum, "Semantic segmentation of mixed crops using deep convolutional neural network," in *Proc. CIGR-AgEng Conf.*, Aarhus, Denmark, Jun. 2016, pp. 1–6.

[38] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Comput. Electron. Agricult.*, vol. 147, pp. 70–90, Apr. 2018.

[39] M. Arsenovic, M. Karanovic, S. Sladojevic, A. Anderla, and D. Stefanovic, "Solving current limitations of deep learning based approaches for plant disease detection," *Symmetry*, vol. 11, no. 7, p. 939, Jul. 2019.

[40] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[41] P. Jiang, Y. Chen, B. Liu, D. He, and C. Liang, "Real-time detection of apple leaf diseases using deep learning approach based on improved convolutional neural networks," *IEEE Access*, vol. 7, pp. 59069–59080, 2019.

[42] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.

[43] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "SSD: Single shot MultiBox detector," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 21–37.

[44] M. Chen, Y. Tang, X. Zou, Z. Huang, H. Zhou, and S. Chen, "3D global mapping of large-scale unstructured orchard integrating eye-in-hand stereo vision and SLAM," *Comput. Electron. Agricult.*, vol. 187, Aug. 2021, Art. no. 106237.

[45] X. Nie, L. Wang, H. Ding, and M. Xu, "Strawberry verticillium wilt detection network based on multi-task learning and attention," *IEEE Access*, vol. 7, pp. 170003–170011, 2019.

[46] Y. Tian, G. Yang, Z. Wang, E. Li, and Z. Liang, "Detection of apple lesions in orchards based on deep learning methods of CycleGAN and YOLOV3-dense," *J. Sensors*, vol. 2019, pp. 1–13, Apr. 2019.

[47] J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," 2018, *arXiv:1804.02767*. [Online]. Available: http://arxiv.org/abs/1804.02767

[48] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4700–4708.

[49] M. Chen, Y. Tang, X. Zou, K. Huang, Z. Huang, H. Zhou, C. Wang, and G. Lian, "Three-dimensional perception of orchard banana central stock enhanced by adaptive multi-vision technology," *Comput. Electron. Agricult.*, vol. 174, Jul. 2020, Art. no. 105508.

[50] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[51] W. Liu, A. Rabinovich, and A. C. Berg, "ParseNet: Looking wider to see better," 2015, *arXiv:1506.04579*. [Online]. Available: http://arxiv.org/abs/1506.04579

[52] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.* Cham, Switzerland: Springer, 2015, pp. 234–241.

[53] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1925–1934.

[54] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters—Improve semantic segmentation by global convolutional network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4353–4361.

[55] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.

[56] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.

[57] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1520–1528.

[58] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 801–818.

[59] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[60] T. Ilyas, A. Khan, M. Umraiz, and H. Kim, "SEEK: A framework of superpixel learning with CNN features for unsupervised segmentation," *Electronics*, vol. 9, no. 3, p. 383, Feb. 2020.

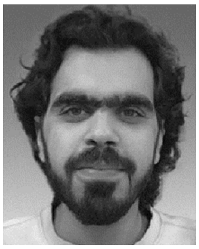
[61] N. Ul Islam and S. Lee, "Interpretation of deep CNN based on learning feature reconstruction with feedback weights," *IEEE Access*, vol. 7, pp. 25195–25208, 2019.

[62] R. A. Rensink, "The dynamic representation of scenes," *Vis. Cognit.*, vol. 7, nos. 1–3, pp. 17–42, 2000.

[63] X. Li, W. Wang, X. Hu, and J. Yang, "Selective kernel networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 510–519.

[64] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: http://arxiv.org/abs/1706.05587

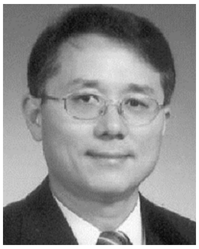
TALHA ILYAS received the bachelor's degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 2017. He is currently pursuing the M.S. degree in electronics and information engineering with Jeonbuk National University, Jeonju, South Korea. His research interests include object instance detection, medical image analysis, object tracking, precision agriculture, optical flow, cam, gan, and depth estimation.



ABBAS KHAN received the bachelor's degree in electrical engineering from Bahria University, Islamabad, Pakistan, in 2018. He is currently pursuing the M.S. degree in electronics and information engineering with Jeonbuk National University, Jeonju, South Korea. His research interests include medical image processing, computer vision, precision agriculture, and depth estimation.

MUHAMMAD UMRAIZ received the bachelor's degree in electrical engineering from COMSATS University Islamabad, Islamabad, Pakistan, in 2017. He is currently pursuing the M.S. degree in electronics and information engineering with Jeonbuk National University, Jeonju, South Korea. His research interests include medical image processing, precision agriculture, and smart farming.

YONGCHAE JEONG (Senior Member, IEEE) received the B.S.E.E., M.S.E.E., and Ph.D. degrees in electronics engineering from Sogang University, Seoul, South Korea, in 1989, 1991, and 1996, respectively. From 1991 to 1998, he worked as a Senior Engineer with Samsung Electronics. In 1998, he joined the Division of Electronics Engineering, Jeonbuk National University, Jeonju, South Korea. From 2006 to 2007, he was with Georgia Institute of Technology, as a Visiting Professor. He is currently a Professor, the Director of the IT Convergence Research Center, and the Director of the HOPE-IT Human Resource Development Center of BK21 PLUS, Chonbuk National University. He has authored or coauthored over 230 papers in international journals and conference proceeding. He is also teaching and conducting research in the area of microwave passive and active circuits, mobile and satellite base-station RF systems, design of periodic defected transmission line, negative group delay circuits and its applications, in-band full duplex radio, and RFIC design. He is a member of Korea Institute of Electromagnetic Engineering and Science (KIEES).



HYONGSUK KIM (Senior Member, IEEE) received the Ph.D. degree in electrical engineering from the University of Missouri, Columbia, MO, USA, in 1992. From 2000 to 2002 and 2009 to 2010, he was a Visiting Scholar with the Nonlinear Electronics Laboratory, Department of Electrical Engineering and Computer Science, University of California at Berkeley, Berkeley, CA, USA. Since 1993, he has been a Professor with the Division of Electronics Engineering, Jeonbuk National University, Jeonju, South Korea. His current research interests include memristors and its application to the implementation of neural networks.