



LS-Net: a convolutional neural network for leaf segmentation of rosette plants

Mainak Deb¹ · Arpan Garai^{2,3} · Arunita Das¹ · Krishna Gopal Dhal¹

Received: 17 August 2021 / Accepted: 26 May 2022 / Published online: 18 June 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract

Leaf segmentation from plant images is a challenging task, especially when multiple leaves are overlapping in images with a complex background. Recently, deep learning-based methods have demonstrated their effectiveness in the realm of image segmentation. In this study, a novel convolutional neural network called LS-Net has been proposed for the leaf segmentation of rosette plants. The experiment is performed over 2010 images from the plant phenotyping (CVPPP) and KOMATSUNA datasets. The segmentation ability of the LS-Net has been investigated by comparing it with four recently applied existing CNN-based segmentation models, namely DeepLab V3 + , Seg Net, Fast-FCN with Pyramid Pooling Module, and U-Net. The analysis of the experimental results clearly demonstrates the superiority of the proposed LS-Net to other tested CNN models.

Keywords Machine learning · Deep learning · Image segmentation · MobilenetV2

1 Introduction

In this era of automation, the agricultural domain requires rigorous manual intervention. This often leads to reduced growth of plants or crops. Therefore, research is going on in the domain of plant phenomics. It refers to the study of the growth and structure of plants. Recently, the extraction of the shape and counting the number of leaves of a plant have become a major issue in plant phenotyping. It is quite

challenging to detect the leaves and count them. It is so because the size of the leaves is quite small and/or the leaves may be in an overlapping manner. Manually counting the leaves of such a plant is a tedious and time-consuming task. Therefore, an automatic process is needed to fix this problem.

Deep learning (DL) is one of the most commonly applied modern artificial intelligence (AI) tools which plays a crucial role in plant phenotyping [1]. Convolution neural networks (CNN) extract the growth dynamics and morphologic traits of an image with greater precision [2]. Recently, plant phenotyping has been recognized as a bottleneck in modern plant breeding and similar research work [3]. Moreover, plant phenotyping is an expanding research field that connects plant biology, technology, DL, and automation engineering. The importance of plant phenotyping is increasing to accelerate the process of plant breeding [4]. Most of the research work in image-based phenotyping is performed based on the structural data and holistic physiological information of the plants present in the image. The phenotypes are extracted using sprouting, appearance, senescence of leaves, and the appearance of fruits and flowers [5]. Related works in phenotyping have increased rapidly over the last few years.

In recent times, digital cameras (often attached to mobile phones) are quite available. Therefore, a robust

✉ Krishna Gopal Dhal
krishnagopal.dhal@midnaporecollege.ac.in

Mainak Deb
mainakdeb15@gmail.com

Arpan Garai
arpangarai@gmail.com

Arunita Das
arunita17@gmail.com

¹ Dept. of Computer Science and Application, Midnapore College (Autonomous), Paschim Medinipur, West Bengal, India

² Dept. of Computer Science and Engineering, University of Engineering and Management, Kolkata, West Bengal, India

³ Dept. of Computer Science and Engineering, Indian Institute of Technology, Delhi, India

plant phenotyping scheme for camera captured images helps to reduce the costs and improve the production of the plants and crops. Therefore, several researchers are investigating plant phenotyping schemes from camera captured images [1]. The growth of a plant can be estimated from the projected leaf area [7]. As mentioned earlier, because the size of the leaves is often small and/or leaves are often overlapped by other leaves, the existing CNN-based plant leaf area segmentation is a challenging task. Furthermore, random rotation or leaf displacement and the lighting effect on plant images make segmentation more difficult [7]. Moreover, the existing CNN-based approaches often fail to produce highly accurate results. Therefore, a novel CNN model has been proposed here to segment the leaf area of plants with higher accuracy. As a result, it can be executed on a system that has limited access to high-performance hardware. It was determined after evaluating previous plant phenotyping studies that there are fewer CNN models with smaller parameter sizes that are associated with good accuracy. CNN models must also be compatible with mobile devices and require less memory. To address this issue, the goal of this research is to design a CNN model with a small number of parameters but high accuracy. The proposed model, LS-Net, has a small number of parameters (approximately 5 M) and requires limited storage. Although the leaf segmentation task is relatively simple for images of rosette plants [3], the existing methods have yet to produce an accurate result. So, the model has been trained and tested using images of rosette plants. The proposed approach is also compared with four different standard CNN models, namely (a) DeepLab V3 + [30]; (b) Seg Net [29]; (c) Fast FCN with Pyramid Pooling module [38]; and (d) U-Net [42]. The leaf segmentation efficacy of the tested CNN models has been investigated by computing segmentation accuracy, dice score, and Intersection over Union (IoU) score. The results are quite encouraging.

In summary, the contributions of the work are as follows:

- A novel convolutional neural network for leaf segmentation of rosette plants has been proposed which consists of a powerful and light-weighted backbone as MobilenetV2, an Atrous Convolution Block which allows the density of encoder features to be controlled [34], and an effective decoder that takes previous information between continuous intervals for better accuracy.
- Unlike existing approaches, a separable convolution has been applied in the Atrous convolution block used for leaf segmentation and in the decoder part, resulting in a quicker and more robust encoder-decoder network.

- On the merged dataset of CVPPP and KOMATSUNA, the proposed model achieves a new state-of-the-art performance. The proposed model was also tested over the CVPPP dataset to perform a better comparison with previously published models.
- A normalization layer has been introduced in this proposed model to filter out the unnecessary pixels from the images. Details are given in Sect. 2.5.

The rest of the paper is organized as follows. Section 1.1 reflects the related existing work. The proposed method is discussed in Sect. 2. The experimental study is presented in Sect. 3. Finally, the concluding remarks are presented in Sect. 4.

1.1 Related works

Successful plant phenotyping research has been carried out in recent years. Several researchers have concentrated their efforts on this hot topic. Phenotyping of plants has been done in climate-controlled laboratories, greenhouses, and even in the outdoors. Many plant phenotyping research projects have been completed recently. Kumar et al. [9] proposed an orthogonal transform domain-based plant region segmentation approach based on orthogonal transform coefficients. To count the leaf counts, they employed a deep CNN. Their models have been tested with mobile phone images and also tested on CVPPP benchmark datasets. With a segmentation accuracy of 93.72%, their recommended system was the most accurate. Wu et al. [10] proposed a highly effective object-aware embedding learning architecture. The architecture includes a module called the distance regression module, which generates seeds for fast clustering. They combined two U-Nets to get an increase in 8% in mean symmetric best dice (mSBD). The fundamental flaw in their suggested model is that it does not produce predicted results for the CVPPP dataset's A3 category. In the leaf segmentation challenge, Gomes and Zheng [11] describe experimental research on the constraints of phenotyping datasets and the best performing approach. They also focused on the notion that model cardinality and test-time augmentation could be useful in single-class image segmentation. Huther et al. [12] have created a versatile pipeline. The approach can extract phenotypic measurements from plant images in an unsupervised manner. Leaf tissue was classified into three categories using segmented images: healthy, anthocyanin-rich, and senescent. Another study had been conducted by Yang et al. [13], where authors investigated how to segment and classify leaf images with a complex background. For this challenge, a Mask R-CNN model had been utilized.

Bell et al. [14] described a method for segmenting *Arabidopsis thaliana* plant images at the leaf level based on identified edges. The leaf margins, as well as the number of leaves, were classified by the model. This method is effective at extracting occluding pairs of leaves that are extensively overlapped. Pape and Kulkas [15] published another paper on leaf segmentation using leaf edge detection. They also offered a strategy that included image analysis based on software IAP [16] for extracting a broad collection of image attributes to forecast the number of leaves. For varied resolution and noise appearance, the suggested model's performance is reduced in the A3 category of the CVPPP dataset. Kuznichov et al. [18] suggested a data augmentation technique that keeps the data objects' geometric structure. The research tried to match the physical appearance of an image as closely as possible to that of authentic images. Researchers got an 86.7 Best-Dice Score by combining different approaches. However, the CVPPP dataset's A2 and A3 categories, on the other hand, had lower segmentation accuracy than A1 and A4.

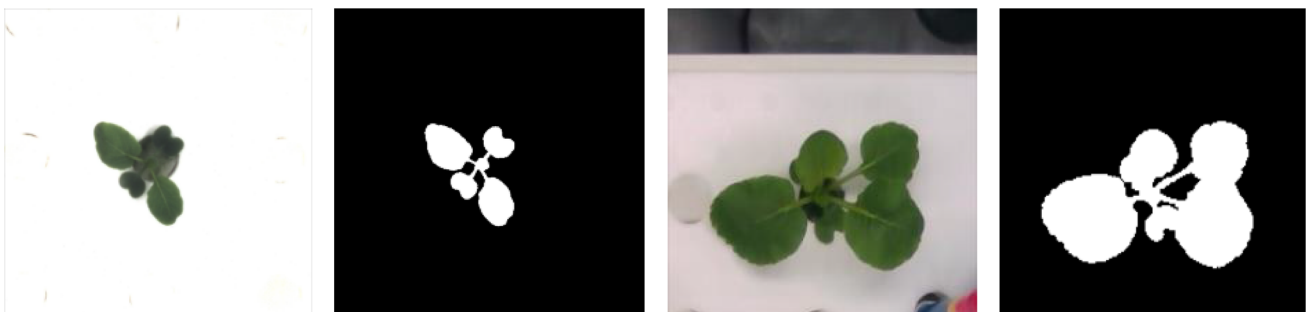
Ubbens et al. [17] used both actual and synthetic datasets to show that the datasets might be interchanged when training a model for the leaf number counting task. They primarily provided a new strategy for complementing plant phenotyping datasets with rendered visuals of synthetic databases. It had been discovered that their strategy reduces the mean absolute count error by 27% approximately.

While most research focused on 2D images, Itakura et al. [21] used an efficient approach to obtain spatial information. The boundaries between all the leaves were unclear in a 2D image of rosette plants because of the overlapped portions. They collected 3D point-cloud images of plants with various cameras and sensors to overcome the drawbacks of a 2D image of the top-view of the plant. As a result, the authors obtained a 0.06 cm² absolute leaf area estimation inaccuracy. Pape and Kulkas [19] provided another 3D histogram-based segmentation for rosette plants where a Euclidean-distance-map-based method had been implemented to detect and segment the leaves. In addition, a method was introduced to detect the optimal leaf split points to separate the overlapped leaves.

Yin et al. [20] suggested a unique paradigm for video processing using fluorescent plants. The number of well-aligned leaves was determined by applying leaf segmentation and alignment to a frame. After that, leaf tracking was used in conjunction with earlier data. The value of the Symmetric Best Dice (SBD) index gained by their proposed model was 78.0. Scharr et al. [22] compared several leaf segmentation methods on a unique dataset containing images from phenotyping experiments. Ren et al. [23] presented an attention-based end-to-end Recurrent Neural Network (RNN) which was trained on RoI (Regions of Interest) that had been successively constructed, with object segmentation inside each region. The suggested model was validated using both the CVPPP and other datasets.



(a) Sample images from CVPPP dataset with their corresponding ground truths



(b) Sample images from KOMATSUNA dataset with their corresponding ground truths

Fig. 1 Example of Leaf images of the utilized datasets

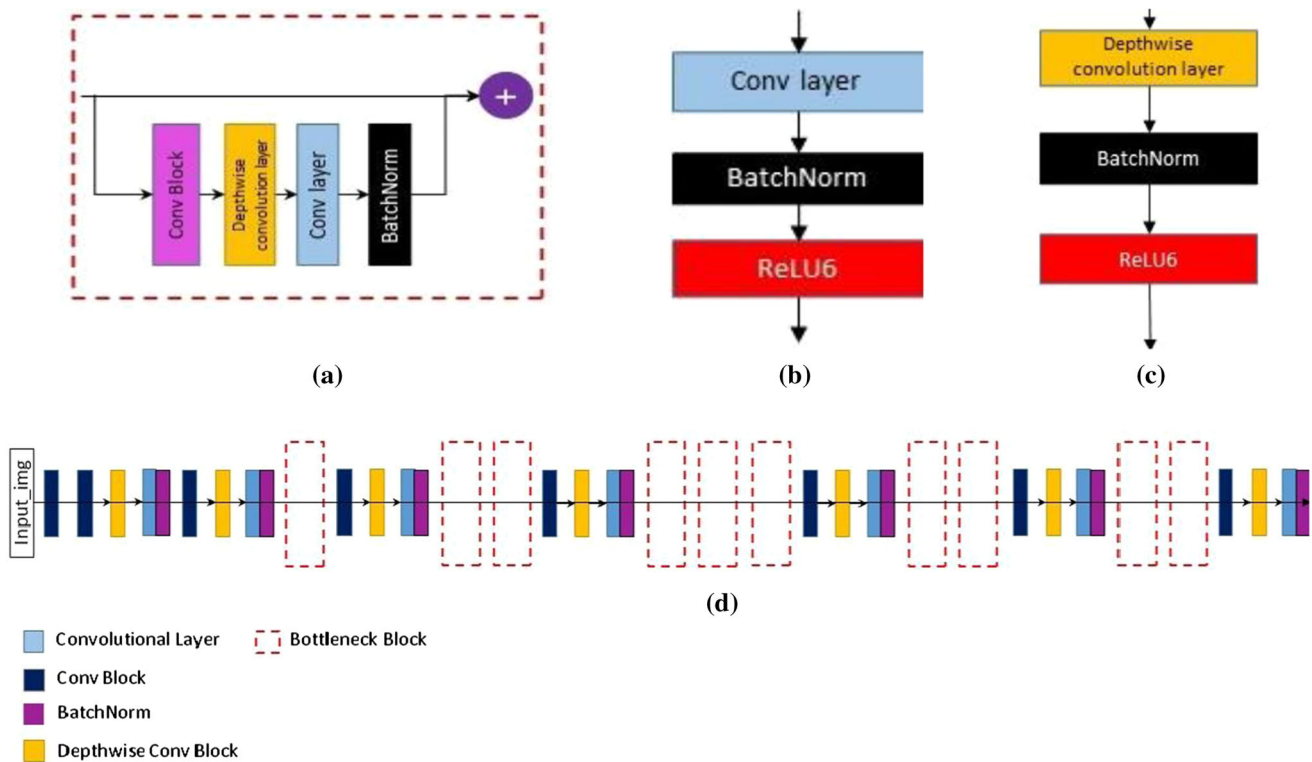


Fig. 2 a Bottleneck Block; b Conv Block; c Depthwise Conv Block; d MobileNetV2 Architecture

Aksoy et al. [24] proposed a multi-level approach for locating and tracking rosette plant leaves. They worked on three weeks old tobacco plants in their early phases of growth with infrared cameras. This allowed for the automatic and non-invasive measurement of essential plant factors such as leaf growth rates. Pre-processing, leaf segmentation, and leaf tracking were the three major parts of this operation. Leaf-shape models were also used to calculate leaf sizes. Dellen et al. [25] evaluated the growth of a tobacco plant using pre-processing time-lapse of growing plants, which was related to plant phenotyping. An image-shape model partitioned the leaf area in each frame. They also developed a new graph-based tracking method that could fully fill gaps in the sequence of a set of surrounding frames. Janssens et al. [26] described yet another unique method for automatic segmentation of individual plant leaves. In addition, an approach for extracting the line of symmetry of the leaf was proposed.

The above discussion clearly demonstrates that accurate leaf segmentation is very important because the accuracy of leaf detection, localization, counting, leaf tracking, and boundary estimation solely depend on this. In addition to that, it is noticed that the majority of the above discussed models are unable to produce satisfactory results over the A2 and A3 categories of the CVPPP dataset due to the green background as noise and a large number of overlapped leaves. Therefore, this study developed a novel

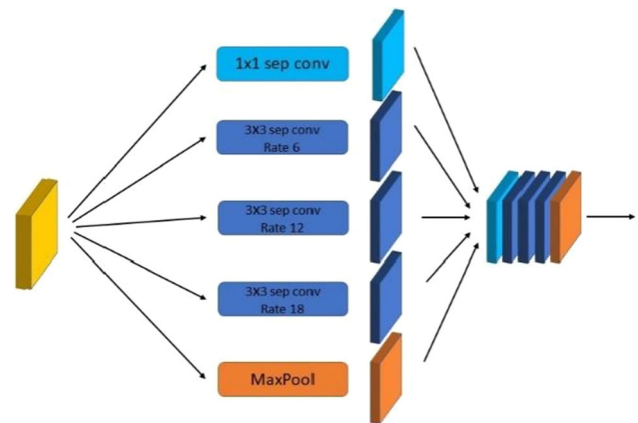


Fig. 3 Atrous Convolution Module

CNN model called LS-Net which is able to perform better leaf segmentation in the presence of various background and overlapping leaves. The brief descriptions of the proposed CNN model and dataset have been done in the following section.

2 Materials and methods

This section describes the proposed CNN model as well as the dataset. The first sub-section describes the used dataset with some sample images. The next sub-section describes

the used backbone of the proposed model. In the next sub-sections, a discussion of Atrous Convolution and Transpose Convolution is presented. In the end, the architecture of the proposed model is demonstrated.

2.1 Dataset design

This experimental study is performed over a merged dataset which is a combination of two well-known datasets, namely the plant phenotyping dataset (CVPPP) benchmark datasets [7] and the KOMATSUNA dataset [8]. The CVPPP dataset is divided into four parts, termed A1, A2, A3, and A4. The number of total images in the four sections is 810.

On the other hand, the KOMATSUNA dataset is divided into two sections. One section is created using an RGB-D camera, and another section is created using multiple RGB cameras. The number of total images in the two sections is 1200. Therefore, there are a total of 2010 images and their segmented ground truths (binary) with the size same as 224×224 in the merged dataset. The images are resized using bilinear interpolation [27]. Subsequently, a random division of the merged dataset has been done to create the train set, validation set, and test set. The training set has 1410 images, the validation set contains 300 images, and the test set contains 300 images, all of which were chosen at random. Random clockwise and anti-clockwise rotations of 90° , 180° , and vertical flips have been performed over images. Some sample images from the dataset are shown in Fig. 1.

A separate dataset of randomly selected 140 images from the CVPPP dataset is utilized for testing in order to perform a better comparison with previously published CNN models.

2.2 Backbone

The CNN backbone determines the precision of an image segmentation model. The feature maps of an image have been extracted by the backbone, and the image has been

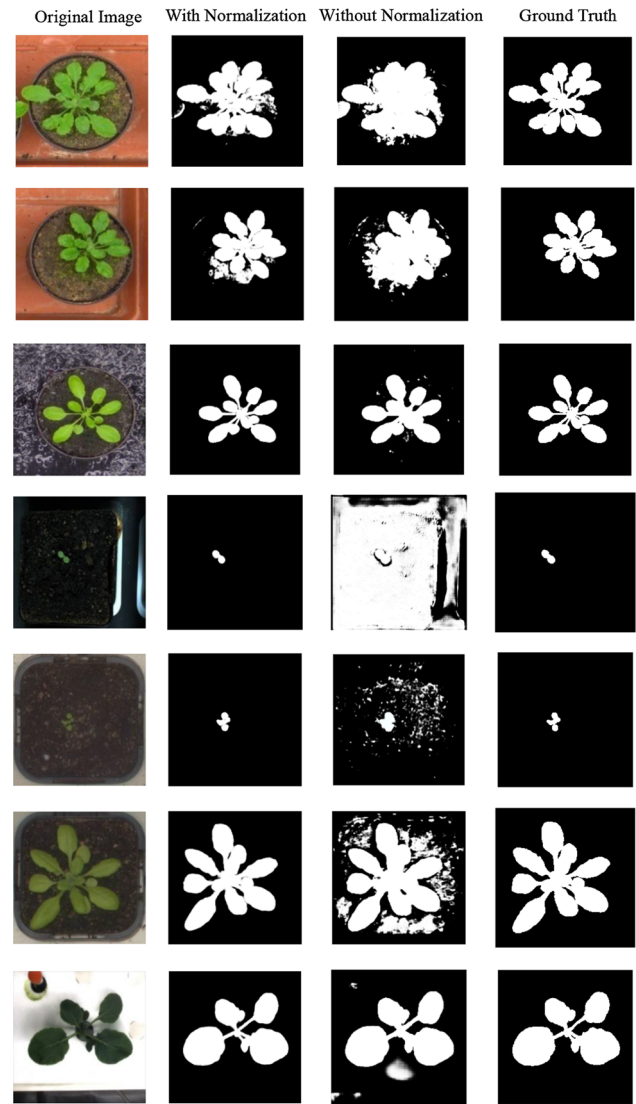
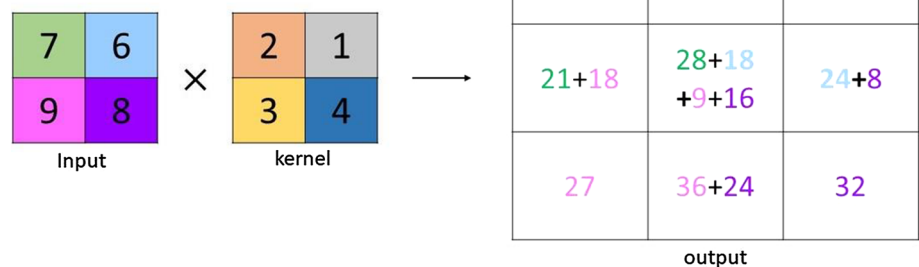


Fig. 5 Segmentation Outcomes of the proposed LS-Net model with and without Normalization Layer

further segmented based on these feature maps. There are many segmentation models with different backbones are reported in [28]. In this paper, MobilenetV2 has been applied as the backbone of the proposed model because it

Fig. 4 Working methodology of Transpose Convolution



proved to be successful in the CNN-based segmentation domain. The MobilenetV2 is discussed in the next paragraph.

The MobilenetV2 [31] model is recently developed based on CNN, which has been utilized by several well-known segmentation models. Hence, it has been applied as a backbone in the proposed model, and the architecture is depicted in Fig. 2. Some minor modifications are made in the architecture of the MobilenetV2 to fit the decoding part. The modifications are discussed in Sect. 4. Many powerful CNN models require the use of depth-wise convolutional layers. This layer reduces both the number of parameters and the amount of computation. A bottleneck block, presented in Fig. 2(a), is employed in MobilenetV2. ReLU6 [31], a special ReLU function, is used here and is defined by the equation $\max(x, 6.0)$. There is a ‘conv block’ consisting of three layers, i.e., a convolution layer, a batch normalization layer, and an activation function as shown in Fig. 2(b). In addition, a ‘depthwise conv block’ which is the combination of depth-wise convolution layer, batch normalization layer, and activation function is presented in Fig. 2(c).

2.3 Atrous convolution block

Atrous convolution block, which is introduced by Chen et. al. [34], has shown efficacy when integrated with well-known DeepLab V3 + [30], DeepLab [35], and PSP Net [36]. Atrous convolutions are very efficient in explicitly controlling the resolution of the feature maps extracted by the deep CNN backbones. For that reason, Atrous Convolution block from Chen et. al. [30] has been adopted for this work. However, the traditional convolution layers are replaced by separable convolution layers because separable convolution layers have significantly fewer parameters than the traditional convolution layers. It also contributes to the reduction of the parameter size in the proposed model. In addition to that, it also sets the field-of-view of the filter in order to capture multi-scale information and simplifies typical convolution operations. In Atrous Convolution, the input is parallelly fed into five layers, namely a (1×1) separable convolution layer, three (3×3) dilated separable convolution layers [37] with dilated rates of 6, 12, and 18, and a Max pooling layer. Then, the outputs of

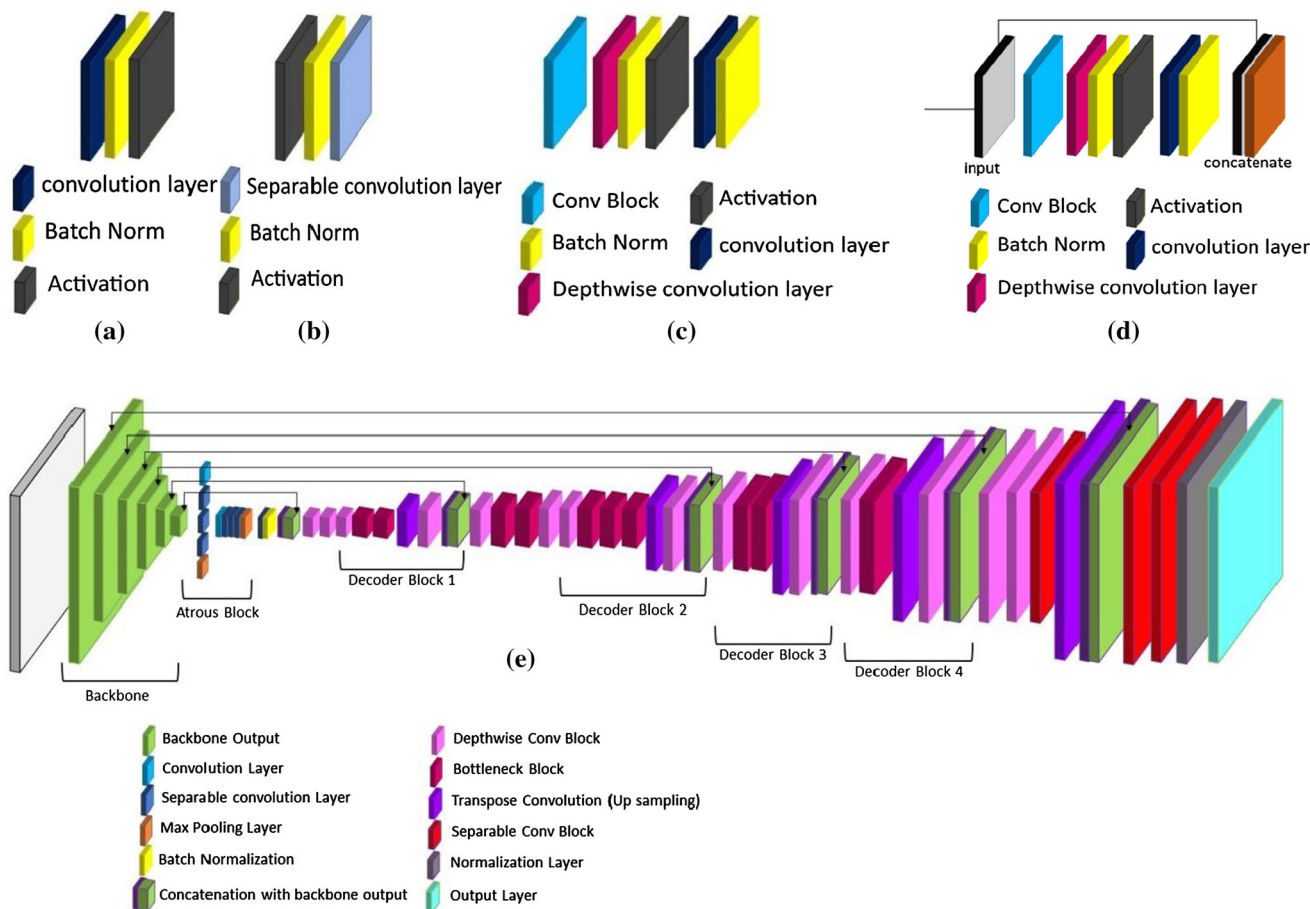


Fig. 6 a Conv Block; b Separable Conv Block; c Depthwise Conv Block; d Bottleneck Block; e Architecture of Proposed LS-Net

all the mentioned layers are concatenated. The architecture is presented in Fig. 3.

2.4 Utilized upsampling method

The upsampling methods, such as nearest neighbour [44], bed of nail [44], max unpooling [44], and bilinear interpolation [27], are predefined and do not depend on data, which makes them task-specific. They do not learn from data and hence are not a generalized technique. To address this issue, a well-known upsampling method called transpose convolution has been applied in this work. Transposed convolutions are the backbone of modern segmentation and super-resolution algorithms. They provide the best and most generalized upsampling of abstract representations. The working methodology is presented in Fig. 4, where it can be easily noticed that output has been generated by performing a point-to-point multiplication between input and kernel [44]. Transposed convolutions also suffer from checkerboard effects, but less than the above-mentioned interpolation techniques. The main cause of this is uneven overlap in some parts of the image, causing artefacts. This can be fixed or reduced by using a kernel-size divisible by the stride, and therefore taking a kernel size of 2×2 or 4×4 when having a stride of 2. Hence, this study utilizes a (2×2) kernel size for transpose convolution. However, for bilinear interpolation, this checkerboard effect exists badly and is significantly higher than transpose convolution.

2.5 Normalization layer

A normalization layer, which has been used in the proposed LS-Net model, is nothing but a layer that makes the values zero that are less than 0.5 and the other values remain unchanged. The idea behind this normalization layer is to filter out the unnecessary pixels from the images. Unwanted pixels can be generated due to the background or the reflection of light. This can decrease the accuracy of the segmentation. With this normalization layer, it has become

easier for the designed model to predict more accurately. Figure 5 shows the two types of segmented images of the model, i.e., with and without normalization layers, and also proves the crucial impact of the normalization layer in the proposed model. The proposed model’s test accuracies with and without the normalization layer are 97.36% and 92.85%, respectively, over the merged dataset. Hence, it can be said that the normalization layer significantly influences the accuracy of the LS-Net. The authors of the other tested CNN models, like DeepLab V3 + , Seg Net, Fast-FCN with Pyramid Pooling Module, and U-Net, did not consider or incorporate the normalization layer into their models. Therefore, this normalization layer is an important component of the proposed LS-Net model as per the experimental study.

2.6 The architecture of the proposed LS-Net Model

The main purpose of proposing this model is to increase the accuracy of the prediction. The proposed LS-Net utilizes MobilenetV2 [31] as the backbone because, besides being efficient, it takes fewer parameters than others. The architecture of our backbone is discussed in the previous section.

Many efficient neural network [31–33] topologies employ depthwise convolution layers as a crucial building element, and this study also uses them. The Separable convolution layers are used to reduce the parameter size. The main concept is to replace a fully convolutional operator with a factorized version that divides convolution into two layers. The first layer, known as depthwise convolution, applies a single convolutional filter per input channel to conduct light-weight filtering. The second layer is a 1×1 convolution, also known as a pointwise convolution, which is responsible for generating new features by calculating linear combinations of the inputs [31].

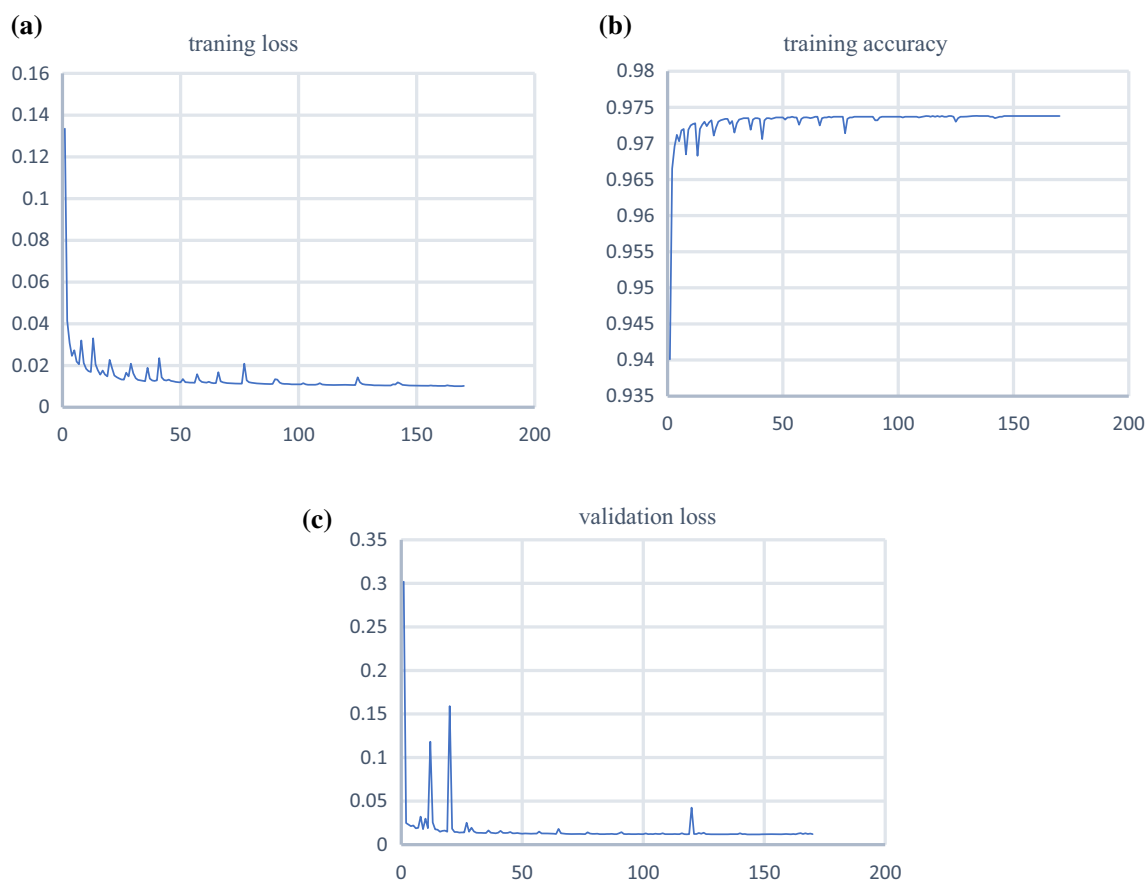
In addition, the image information has been saved before each downsampling and used later in the model, as shown in Fig. 6. ReLU6 [31] has been used as the

Table 1 Segmentation Quality Parameters

Sl	Parameters	Formulation	Remarks
1	Accuracy (AC)	$AC = \frac{TP+TN}{TP+TN+FP+FN}$	It is defined as the sum of the correctly classified pixels divided by the sum of the total number of pixels. A high AC value indicates a better result [39]
2	Dice score (DI)	$DI = \frac{2 \times A \cap B }{ A + B }$	Dice measures how much area of the predicted image matched with the ground truth. In other words, it calculates overlapped area between the predicted image and the ground truth image [40]. A high DI value indicates a better result [40]
3	Intersection over Union (IoU)	$IoU = \frac{ A \cap B }{ A \cup B }$	It is a score between 0 and 1 that shows how much area is overlapped between the predicted image and ground truth. If the score is above 0.5 then the model predicts well [41]

Table 2 Parameter Settings of CNN models over input image size: (224, 224, 3)

CNN model	Parameter setting
DeepLab V3 +	The filters of the backbone are the same as the original. The sigmoid activation function is used for prediction. Total 3.5 M trainable parameters exist for DeepLab V3 + [30]
Seg Net	The filters of the backbone and decoder part are the same as the original. At last, the sigmoid activation function is used for prediction. the Seg Net model has 16.7 M trainable parameters [29]
Fast FCN with Pyramid Pooling module	The filters of the backbone and decoder part are the same as the original. At last, the sigmoid activation function is used for prediction. Fast FCN model associates with 51.8 M trainable parameters [38]
U-Net	The filters of the backbone and decoder part are the same as the original. At last, the sigmoid activation function is used for prediction. U-Net model has 31.0 M trainable parameters [42]
Proposed LS-Net	The batch size for the proposed model is 7. The filters of the backbone are the same as the original. For Atrous convolution block, a filter of size 128 has been used. In the next separable conv layer, proposed model uses 864 filters. After the first concatenation, the number of filters is grown to 1024, and subsequently the number of filters is gradually reduced. Finally, the sigmoid activation function is employed in the prediction process. The proposed model has 5 million trainable parameters

**Fig. 7** Training details of Proposed LS-Net: **a** training loss; **b** training accuracy; **c** validation loss

activation function. The Atrous convolution block, as presented in Fig. 3 is added after the backbone. The entire architecture is illustrated in Fig. 6. The kernel size of the separable convolution layer is (1×1) . The kernel size of all the ‘separable conv block’ is (3×3) .

In the decoder part, there are four decoder blocks and consists of one depthwise conv block as presented in Fig. 6(c), two bottleneck blocks as presented in Fig. 6(d), a transpose convolution as shown in Fig. 4 [44], and a depthwise conv block, and concatenation.

Table 3 Training results of the Proposed LS-Net

Models	Parameters (trainable)	MS/STEP	Epoch no	Training accuracy	Validati on loss	Training loss
Proposed ls-net model	5.0 M	950 ms	1	94.01%	0.3019	0.1335
			100	97.37%	0.0121	0.0110
			170	97.38%	0.0120	0.0102

Table 4 The testing results obtained by different CNN model over merged dataset

Models	Test loss	Test accuracy	Dice score	IoU
DeepLab V3 +	0.0223	97.13%	94.59	89.78
Seg net	0.0176	97.29%	95.04	90.58
Fast FCN with pyramid pooling block	0.0194	97.24%	94.31	89.29
U-Net	0.0580	96.40%	91.87	86.50
Proposed LS-net	0.0112	97.36%	96.70	93.61

*Best results are marked in bold

A depthwise conv block as shown in Fig. 6(c) consists of a conv block as presented in Fig. 6(a) with (1×1) kernel size, a (3×3) depthwise convolution layer [31] followed a batch normalization and an activation function, a (1×1) convolution layer [39], and a batch normalization layer.

The bottleneck block is the same as the depthwise conv block but the only difference is that, at the end, the input is concatenated with the output of that block which is presented in Fig. 6(d). A bottleneck block consists of a conv block with (1×1) kernel size, a (3×3) depthwise convolution layer [31] followed by batch normalization and an activation function, a (1×1) convolution layer [39], and a batch normalization layer, and concatenation.

3 Experimental results

The experimental study has been performed using NVIDIA GeForce 1650 with cuDNN CUDA 10.0 and AMD Ryzen 5 3550H processor with 16 GB RAM and 256 GB SSD. On the other hand, the software is Anaconda, which includes Jupyter Notebook. All experiments have been performed using TensorFlow and Scikitlearn.

The performance measures of segmentation work have been investigated by computing three widely used parameters, namely segmentation accuracy, dice score, and intersection over union (IoU). The details of the parameters are reported in Table 1.

3.1 Parameter settings

The ‘Binary Crossentropy’ has been utilized as the loss function for all CNN models. The input image is in RGB format and the shape is $(224, 224, 3)$. The output image is in binary form and of size $(224, 224, 1)$. The batch size varies according to the proposed model and machine capacity. The parameter settings of the proposed and other tested CNN models are presented in Table 2.

3.2 Training and validation of the proposed model

Images that belong to the training and validation sets are normalized before training and preprocessing. In the proposed model, the Adam optimizer and binary crossentropy as the loss function have been utilized. To get the optimized result, the training process has been terminated when the training loss is less than or equals 0.011, which is set empirically. The training accuracy began at 94.01% and ended at 97.38%, while the training loss began at 0.1335 and ended at 0.0102. The training details of the proposed LS-Net are presented in Fig. 7. The training outcomes, on the other hand, are shown in. Table 3.

3.3 Discussion on testing and prediction

The segmentation efficacy of the tested CNN models has been measured by computing the Dice score and IoU along with testing loss and accuracy. The proposed LS-Net model has been compared with DeepLab V3 +, Seg Net, Fast FCN with Pyramid Pooling module, U-Net over merged

Fig. 8 Graphical analysis of Dice score and IoU: **a** DeepLab V3 + ; **b** Seg Net; **c** Fast-FCN with Pyramid Pooling Module; **d** U-Net; **e** Proposed LS-Net Model

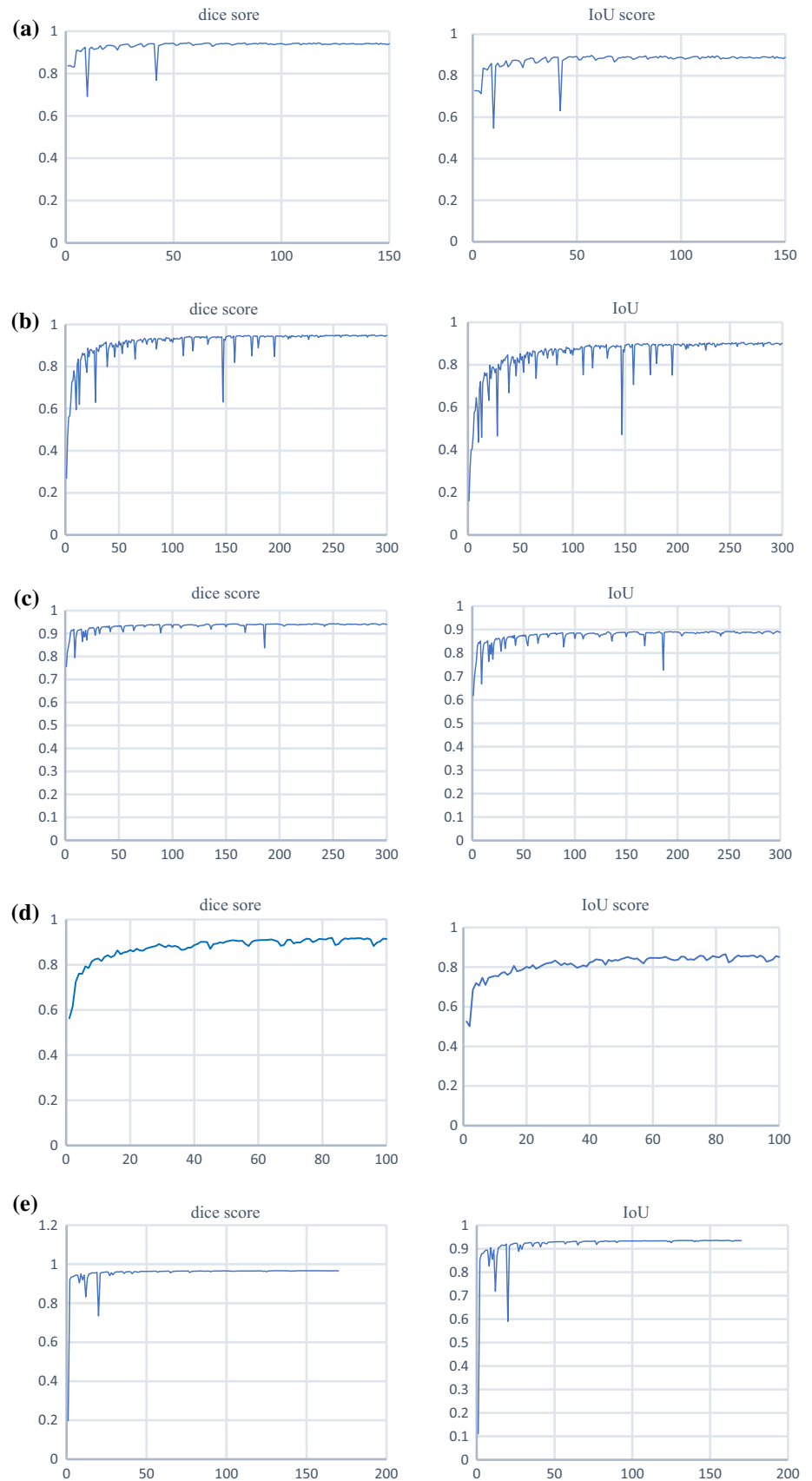


Table 5 The testing results obtained by different CNN model over CVPPP Dataset

Models	Test loss	Test accuracy	Dice score	IoU
Deeplab V3 +	0.0237	98.21%	91.79	85.11
Seg net	0.0197	97.92%	92.50	86.04
Fast FCN with pyramid pooling block	0.0211	97.57%	92.54	85.94
U-Net	0.0598	96.92%	89.23	81.55
Proposed LS-net	0.0118	98.92%	96.51	93.30

*Best results are marked in bold

Table 6 Dice Score comparison with prior published leaf segmentation models for leaf segmentation on CVPPP Dataset

Nottingham [9]	Wageningen [9]	Kumar & Domnic [9]	Kumar & Domnic [45]	SLIC_Seg [22, 45]	Proposed LS-Net
91.6	94.0	94.7	95.4	91.2	96.51

*Best results are marked in bold

dataset, and CVPPP dataset. In addition to that, LS-Net is also compared with SLIC_Seg [22, 45], a segmentation model proposed by Nottingham [9], Wageningen [9], Kumar & Domnic [9], Kumar & Domnic [45] over the CVPPP dataset based on Dice score.

The numerical values of the mentioned quality measurement parameters over the merged dataset have been recorded in Table 4. In addition to that, the graphical analysis of the dice score and IoU for all employed models over the merged dataset is performed in Fig. 8. The results over the CVPPP dataset are presented in Tables 5 and 6, whereas the segmented leaf outcomes by tested CNN models are presented in Fig. 9.

According to Table 4 and the graphical analysis of Fig. 8, it is clear that the proposed LS-Net model delivers superior results compared to other tested CNN models in terms of test accuracy, test loss, Dice, and IoU over the merged dataset. Seg Net provided second best results when the same evaluation parameters have been considered. DeepLab V3 + and Fast FCN with Pyramid Pooling block produced nearly the same results. U-Net, on the other hand, produced the worst results.

The tested CNN models were evaluated again on the CVPPP dataset. The numerical values of test accuracy, test loss, Dice, and IoU in Table 5 clearly show that the proposed LS-Net produced competitive results when compared to others. LS-Net achieved a testing accuracy of 98.92%. On the other hand, finding the second best model over the CVPPP dataset is quite difficult. DeepLab V3 + is the second best model based solely on testing accuracy. When test loss and IoU were taken into account, Seg Net produced the second best result. Fast FCN with Pyramid Pooling block provided a slightly better Dice score compared to Seg Net. U-Net also produced the worst numerical results in this case. Table 6 compares the proposed LS-Net

model to some state-of-the-art segmentation techniques by computing Dice scores over the CVPPP datasets. The Dice score adds credence to the superiority of the LS-Net. The dice scores of the compared state-of-the-art techniques over CVPPP dataset are collected from [9, 45].

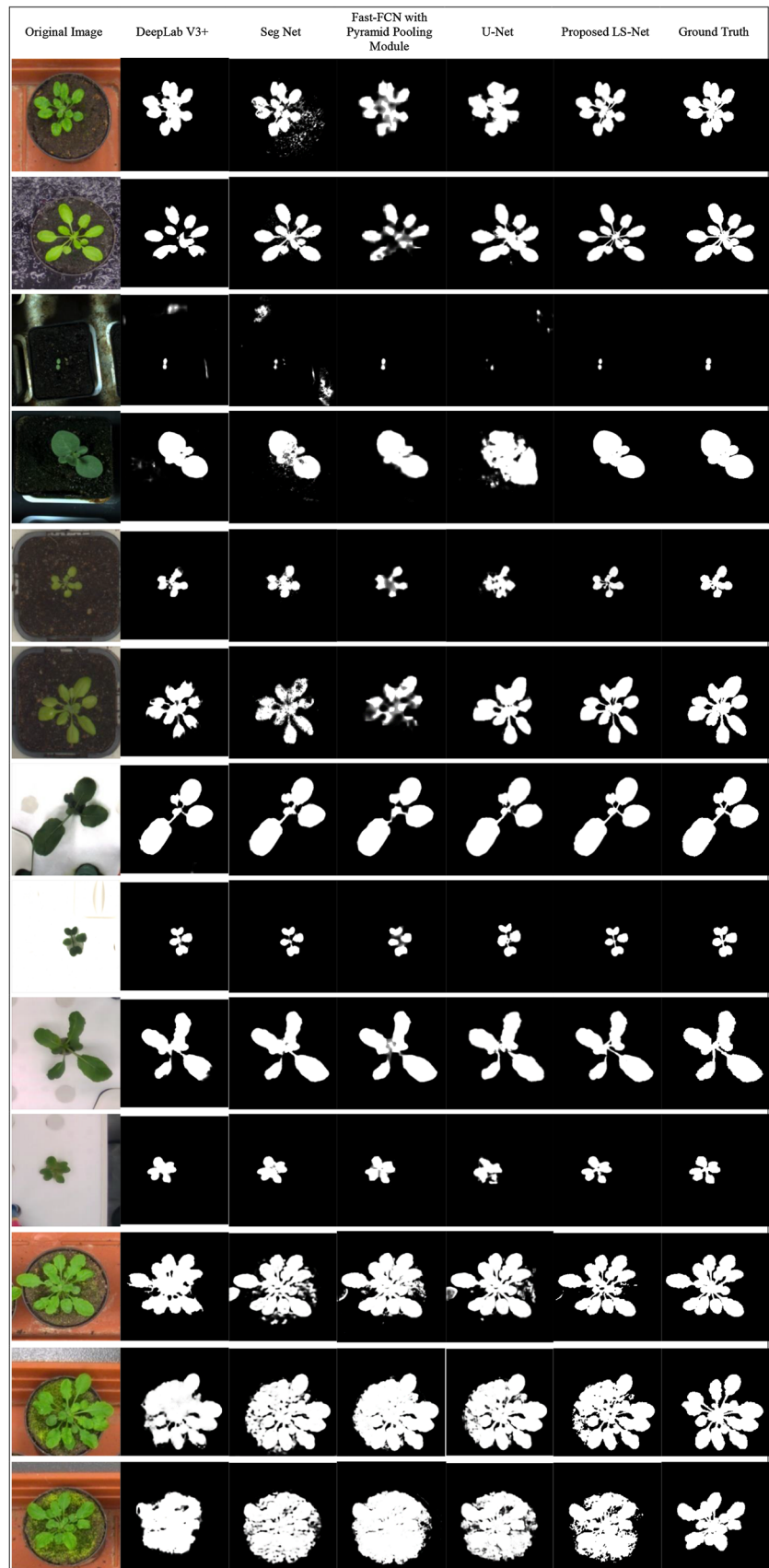
The leaf segmented outcomes of the tested CNN models over the merged dataset are presented in Fig. 9 for visual analysis following the quality parameter-based analysis. The ground truth images of the tested images are also included in Fig. 9 for better visual comparison. Overall, the proposed LS-Net produces better segmented results than other tested CNN models. Segmented results over complex images are also presented for better visual comparison of the tested CNN models. Images with a small green background are considered as complex because the colours of the background and the leaf are very similar. The last three rows of Fig. 9 show the prediction results of the models used, demonstrating that LS-Net produced competitive results.

4 Conclusion

This paper introduces LS-Net, a Deep Convolution Neural Network (DCNN) for the leaf segmentation of rosette plants. The LS-Net is trained and evaluated on the CVPPP [7] and KOMATSUNA [8] datasets. DeepLab V3 +, Seg Net, Fast-FCN with Pyramid Pooling Module, and U-Net are four well-known DCNN models that are compared to the proposed LS-Net to investigate the efficacy of the proposed one. Depending on the architecture and experimental results, the following conclusions can be made:

1. The proposed LS-Net model is better in the leaf segmentation domain because of its powerful and light-

Fig. 9 Segmented outcomes of the tested CNN Models



weighted backbone, i.e., MobilenetV2, an Atrous convolution block that allows the density of encoder features to be controlled, and an effective decoder that takes previous information between continuous intervals.

2. The utilization of normalization layer in the proposed LS-Net model has a great impact because it helps to filter out the unnecessary pixels from the images. The LS-Net also takes up less memory and therefore is more suited than the other models reviewed.
3. The visual and numerical findings clearly show that the LS-Net produces competitive outcomes in the leaf segmentation area when compared to other state-of-the-art CNN models. The key advantage of the proposed LS-Net model over other evaluated CNN models is that it can better segment complex leaf images, such as those with a light green background.
4. Over the merge dataset and the CVPPP dataset, the LS-Net obtains 97.36% and 98.92% test accuracies, respectively, demonstrating its effectiveness in the field of leaf segmentation. This is supported not only by accuracy, but also by other well-known segmentation quality scores such as dice and IoU.

However, the limitation of the proposed model is that it provides unsatisfactory results when there is a large overlap between leaves or images have a significant green background. Based on the above conclusions and limitations, the following future directions for the study are as follows:

1. Although LS-Net performs well in the leaf segmentation of rosette plants, it is important to test it in other real-world leaf segmentation areas as well as medical image segmentation areas.
2. Accurate leaf segmentation is required for leaf counting. As a result, leaf counting using an LS-Net-based segmentation model could be a promising future endeavour.
3. Aside from the development of CNN models, enhancing image quality with better advanced techniques can also help with performance. Furthermore, utilizing an embedding system, it will be feasible to segment and classify leaves automatically in real-time.
4. Finally, but certainly not least, deep learning seems to have its own set of common issues, such as network structure design, 3D image segmentation model design, and loss function design. Designing 3D convolution models to analyze 3D leaf image data is a researchable direction. Loss function design has long been a challenge in deep learning research. Nature-inspired optimization algorithms [46] based optimized deep learning models [47] can also be an emerging research topic.

Author contributions MD contributed to conceptualization, methodology, and software. AG contributed to visualization, investigation, and validation. AD involved to writing—original draft preparation. KGD involved in supervision and writing—reviewing and editing.

Funding There is no funding associated with this research.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest. The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Chandra AL, Desai SV, Guo W, and Balasubramanian VN (2020) Computer vision with deep learning for plant phenotyping in agriculture: a survey. arXiv preprint [arXiv:2006.11391](https://arxiv.org/abs/2006.11391).
2. Jiang Y, Li C (2020) Convolutional neural networks for image based high throughput plant phenotyping a review. *Plant Phenomics*. <https://doi.org/10.34133/2020/4152816>
3. Furbank RT, Tester M (2011) Phenomics—technologies to relieve the phenotyping bottleneck. *Trends Plant Sci* 16(12):635–644
4. Fiorani F, Schurr U (2013) Future scenarios for plant phenotyping. *Annu Rev Plant Biol* 64:267–291
5. Das Choudhury S, Samal A, Awada T (2019) Leveraging image analysis for high-throughput plant phenotyping. *Front Plant Sci* 10:508
6. Zhang C, Marzougui A, Sankaran S (2020) High-resolution satellite imagery applications in crop phenotyping: an overview. *Comput Electron Agric* 175:105584
7. Minervini M, Fischbach A, Scharr H, Tsaftaris SA (2016) Finely-grained annotated datasets for image-based plant phenotyping. *Pattern Recogn Lett* 81:80–89
8. Uchiyama H, Sakurai S, Mishima M, Arita D, Okayasu T, Shimada A, and Taniguchi RI (2017) An easy-to-setup 3D phenotyping platform for KOMATSUNA dataset. In: *Proceedings of the IEEE international conference on computer vision workshops* (pp. 2038–2045)
9. Kumar JP, Domnic S (2020) Rosette plant segmentation with leaf count using orthogonal transform and deep convolutional neural network. *Mach Vis Appl* 31(1):1–14
10. Wu Y, Chen L, and Merhof D (2020, August). Improving pixel embedding learning through intermediate distance regression supervision for instance segmentation. In: *European conference on computer vision*. Springer, Cham, (pp. 213–227)
11. Gomes DPS, and Zheng L (2020) Leaf segmentation and counting with deep learning: on model certainty, Test-time augmentation, Trade-Offs. arXiv preprint [arXiv:2012.11486](https://arxiv.org/abs/2012.11486).
12. Hüther P, Schandry N, Jandrasits K, Bezrukov I, and Becker C (2020) ARADEEPOPSIS, an automated workflow for top-view plant phenomics using semantic segmentation of leaf states. *The Plant Cell*. <https://doi.org/10.1105/tpc.20.00318>
13. Yang K, Zhong W, Li F (2020) Leaf segmentation and classification with a complicated background using deep learning. *Agronomy* 10(11):1721

14. Bell J, and Dee HM (2019) Leaf segmentation through the classification of edges. arXiv preprint [arXiv:1904.03124](https://arxiv.org/abs/1904.03124).
15. Pape JM, and Klukas C (2015) Utilizing machine learning approaches to improve the prediction of leaf counts and individual leaf segmentation of rosette plant images. In: Proceedings of the Computer Vision Problems in Plant Phenotyping (CVPPP), 1–12
16. Klukas C, Chen D, Pape JM (2014) Integrated analysis platform: an open-source information system for high-throughput plant phenotyping. *Plant Phys* 165(2):506–518
17. Ubbens J, Cieslak M, Prusinkiewicz P, Stavness I (2018) The use of plant models in deep learning: an application to leaf counting in rosette plants. *Plant Methods* 14(1):1–10
18. Kuznichov D, Zvirin A, Honen Y, and Kimmel R (2019) Data augmentation for leaf segmentation and counting tasks in rosette plants. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (pp. 0–0)
19. Pape JM, and Klukas C (2014, September) 3-D histogram-based segmentation and leaf detection for rosette plants. In: European conference on computer vision, Springer, Cham, (pp. 61–74)
20. Yin X, Liu X, Chen J, Kramer DM (2017) Joint multi-leaf segmentation, alignment, and tracking for fluorescence plant videos. *IEEE Trans Pattern Anal Mach Intell* 40(6):1411–1423
21. Itakura K, Hosoi F (2018) Automatic leaf segmentation for estimating leaf area and leaf inclination angle in 3D plant images. *Sensors* 18(10):3576
22. Scharr H, Minervini M, French AP, Klukas C, Kramer DM, Liu X, Tsaftaris SA (2016) Leaf segmentation in plant phenotyping: a collation study. *Mach Vis Appl* 27(4):585–606
23. Ren M, and Zemel RS (2017) End-to-end instance segmentation with recurrent attention. In: Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 6656–6664)
24. Aksoy EE, Abramov A, Wörgötter F, Scharr H, Fischbach A, Dellen B (2015) Modeling leaf growth of rosette plants using infrared stereo image sequences. *Comput Electron Agric* 110:78–90
25. Dellen B, Scharr H, Torras C (2015) Growth signatures of rosette plants from time-lapse video. *IEEE/ACM Trans Comput Biol Bioinf* 12(6):1470–1478
26. Janssens O, De Vylder J, Aelterman J, Verstockt S, Philips W, Van Der Straeten D, and Van de Walle R (2013, September) Leaf segmentation and parallel phenotyping for the analysis of gene networks in plants. In: 21st European signal processing conference (EUSIPCO 2013). IEEE (pp. 1–5)
27. Patel V, Mistree K (2013) A review on different image interpolation techniques for image enhancement. *Int J Emerg Technol Adv Eng* 3(12):129–133
28. Minaee S, Boykov YY, Porikli F, Plaza AJ, Kehtarnavaz N, and Terzopoulos D (2021) Image segmentation using deep learning: a survey. In: IEEE transactions on pattern analysis and machine intelligence.
29. Badrinarayanan V, Kendall A, Cipolla R (2017) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans Pattern Anal Mach Intell* 39(12):2481–2495
30. Chen LC, Zhu Y, Papandreou G, Schroff F, and Adam H (2018) Encoder-decoder with Atrous separable convolution for semantic image segmentation. In: Proceedings of the European conference on computer vision (ECCV) (pp. 801–818)
31. Sandler M, Howard A, Zhu M, Zhmoginov A, and Chen LC (2018) Mobilenetv2: Inverted residuals and linear bottlenecks. In: Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 4510–4520).
32. Howard AG, Zhu M, Chen B, Kalenichenko D, Wang W, Weyand T, and Adam H (2017) Mobilenets: Efficient convolutional neural networks for mobile vision applications. arXiv preprint [arXiv:1704.04861](https://arxiv.org/abs/1704.04861).
33. Chollet F (2017) Xception: Deep learning with depthwise separable convolutions. In: Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1251–1258)
34. Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2017) Deeplab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans Pattern Anal Mach Intell* 40(4):834–848
35. Chen LC, Papandreou G, Kokkinos I, Murphy K, and Yuille AL (2014) Semantic image segmentation with deep convolutional nets and fully connected crfs. arXiv preprint [arXiv:1412.7062](https://arxiv.org/abs/1412.7062).
36. Zhao H, Shi J, Qi X, Wang X, and Jia J (2017) Pyramid scene parsing network. In: Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 2881–2890)
37. Yu F, and Koltun V (2015) Multi-scale context aggregation by dilated convolutions. arXiv preprint [arXiv:1511.07122](https://arxiv.org/abs/1511.07122).
38. Wu H, Zhang J, Huang K, Liang K, and Yu Y (2019) Fastfcn: rethinking dilated convolution in the backbone for semantic segmentation. arXiv preprint [arXiv:1903.11816](https://arxiv.org/abs/1903.11816).
39. Khan A, Sohail A, Zahoora U, Qureshi AS (2020) A survey of the recent architectures of deep convolutional neural networks. *Artif Intell Rev* 53(8):5455–5516
40. Sudre CH, Li W, Vercauteren T, Ourselin S, Cardoso MJ (2017) Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. *Deep learning in medical image analysis and multimodal learning for clinical*. Springer, Cham, pp 240–248
41. Rahman MA, and Wang Y (2016, December) Optimizing intersection-over-union in deep neural networks for image segmentation. In: International symposium on visual computing. Springer, Cham, (pp. 234–244)
42. Ronneberger O, Fischer P, and Brox T (2015, October) U-net: Convolutional networks for biomedical image segmentation. In: International Conference on Medical image computing and computer-assisted intervention. Springer, Cham, (pp. 234–241)
43. Santurkar S, Tsipras D, Ilyas A, and Madry A (2018) How does batch normalization help optimization?. arXiv preprint [arXiv:1805.11604](https://arxiv.org/abs/1805.11604).
44. Jeremy Jordan: An overview of semantic image segmentation. <https://www.jeremyjordan.me/semantic-segmentation>. Access Date: 19.04.2021
45. Kumar JP, Domnic S (2019) Image based leaf segmentation and counting in rosette plants. *Inf Process Agric* 6(2):233–246
46. Dhal KG, Das A, Ray S, Gálvez J, Das S (2020) Nature-inspired optimization algorithms and their application in multi-thresholding image segmentation. *Arch Comput Methods Eng* 27(3):855–888. <https://doi.org/10.1007/s11831-019-09334-y>
47. Vrbančić G, Fister Jr I, and Podgorelec V (2018, June) Swarm intelligence approaches for parameter setting of deep learning neural network: case study on phishing websites classification. In: Proceedings of the 8th international conference on web intelligence, mining and semantics (pp. 1–8)

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