IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received March 12, 2021, accepted March 22, 2021, date of publication April 16, 2021, date of current version April 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3073715

End to End Segmentation of Canola Field Images Using Dilated U-Net

HAFIZ SAMI ULLAH, MUHAMMAD HAMZA ASAD, AND ABDUL BAIS, (Senior Member, IEEE)
Faculty of Engineering and Applied Science, University of Regina, Regina, SK S4S 0A2, Canada

Corresponding author: Abdul Bais (abdul.bais@uregina.ca)

This work was supported by the Natural Sciences and Engineering Research Council of Canada Alliance Grant entitled Ground Truth Validation of Crop Growth Cycle Using High Resolution Proximal and Remote Sensing with Phantom Ag Ltd., and in part by the Croptimistic Technology Inc.

ABSTRACT Semantic segmentation is used in many fields like agriculture, medical imaging, and autonomous driving. The paper proposes an end to end solution for efficient weeds and crop segmentation in field environment application. The crop/weeds segmented output is utilized to generate a decision map for variable rate fertilizer and herbicide application. Currently available models are memory expensive and do not have real time performance unless enough computational power is accessible in field. We use Maximum Likelihood Classification (MLC) and image processing techniques to label field images in three classes; background, crop, and weeds. This data is processed through our modified U-Net, which improves the semantic accuracy with reduced memory cost. We train our model with DICE loss and compare the results with state of the art. We achieve 89.12% mean Intersection Over Union (mIOU) with 86.11%, 82.99%, and 98.23% individual IOU for crop, weeds, and background, respectively. Our proposed model uses only 15M parameters which are 57M less than the state-of-the-art models with a compromise of 1% mIOU score.

INDEX TERMS Canola detection, weed control, maximum likelihood classification, dilated convolution, depth wise separable convolution.

I. INTRODUCTION

Semantic segmentation is the understanding and classification of image regions at the pixel level. It helps in getting information about the objects in an image, and their estimated locations. It is an important task in computer vision to know object boundaries for precision applications. Deep Convolutional Neural Networks (DCNN) have made it possible to solve such high level problems. DCNNs like AlexNet [1], VGG [2], ResNet [3], and Inception [4] are well known architectures. These networks were originally proposed for image classification but now they are also being used for related tasks like translation and semantic mapping. These networks extract deep features in an image to understand its content. The information we get from semantic mapping is helpful in many fields like autonomous driving [5]–[8], medical imaging [9], [10], and precision agriculture [11]–[13]. In this paper, we use semantic segmentation to detect weeds and crop in canola field images. The detection of weeds and crop helps in generating a map of the field which is used in variable rate herbicides application.

One of the challenging tasks of performing semantic segmentation is image labelling. In each field, type of crop, weeds, and soil texture are changing. Moreover, the shape of plant is changing with growth. These constraints make it difficult to provide a general solution towards weeds and crop mapping. For each crop and soil type we may need to label data from scratch. Previously published dataset on sugar beet crop for weeds detection, is more of an experimental data [14]. With constant light, high definition, and multi-spectral sensors the dataset simplifies the real field environment challenges. It is not economical and scalable to collect data under such constraints. Our targeted dataset is collected using commercially available simple RGB camera. The images are taken in uncontrolled environment and it presents the real-field problem. While labelling the images, equipment vibration caused blurring. The last season dead plants and plant shadow posed significant challenge. Under the given circumstances, vegetation extraction to remove the background becomes extremely difficult.

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

Considering the challenges of uncontrolled field conditions data and less effective output of vegetation extracted through manual indices, we use a semi-automatic and robust procedure to segment background pixels. The extracted vegetation is fine tuned and processed in several steps to make labelling of images time efficient. We segment background using Maximum Likelihood Classification (MLC) [15]. Once the background is subtracted, we use noise removing techniques to fine tune the extracted vegetation. After extracting the vegetation, we add a human annotator in the loop to label the minority class pixels. Once we have vegetation and one class labelled (either crop or weed), we use image subtraction to get other class labelled. In the final step, noise is removed and missing pixels are labelled.

For weeds/crop segmentation task, we need a solution which is cost effective in terms of computational and memory resources for edge deployment. It should also perform at par with the state of the art. The state of the art models have high computational and memory requirements. We propose a modified version of U-Net [10] for the semantic segmentation. To improve segmentation, we pass the input image through three parallel Dilated Convolutional Layers (DCLs). This process helps in the extraction of relevant cues like boundary information and object connectivity which are important in the segmentation task. These cues placge in semantic translation. Once we have the relevant features, they are processed using traditional model but with improved accuracy. We perform our experiments with ResNet-50 [3] as encoder and use U-Net [10], SegNet [16], HRNet_Mscale [17], and Deeplabv3+ [18] as decoder architectures. For training the models, we use DICE loss to measure the overlap. To quantify the improvements, a brief comparison is made. We also perform computational cost comparison to describe the effectiveness of the proposed modifications.

The major contributions of this work are listed below:

- A complete end-to-end solution for agricultural image segmentation which includes semi-automated image labelling for three class problem and crop/weeds segmentation for a real-world agriculture data.
- An efficient light weight network that helps in mitigating memory and performance issues for remote deployment.

The paper is arranged in 5 sections. In Section 2 we review the related literature, Section 3 covers the labelling process and model modifications. We also include ablation study in Section 3. We discuss results and make comparison between different models in Section 4. We conclude our work and discuss the recommendations for future work in Section 5.

II. RELATED WORK

Semantic segmentation is an active research area. Most of the research in this domain is related to autonomous driving. The state of the art models use multiple streams to process information like shape stream (residual shape processing blocks) and regular stream (backbone architecture) [19]. Some of the researchers use feature selection mechanism based on Gated Convolutional Layers (GCLs) [20] to improve segmentation. We provide literature review related to techniques focused on improving segmentation.

Liu *et al.* [21] study the Neural Architecture Search (NAS) for image segmentation. The search is performed on network and cell levels yielding promising results for dense image prediction. The NAS based design exceeds human proposed networks given the large data. Stochastic gradient descent is used for hierarchical architecture search. To target the computational cost, constraint differentiable formulation of NAS [22], [23] is used. They achieve 85.6% mean Intersection Over Union (mIOU) on the COCO dataset without the use of pre-trained models.

DeepLab [24] uses DCL in which the filters are padded to receive larger field. It helps in maintaining feature map with higher resolution. Networks like Deeplabv3+ [18] employ Atrous Spatial Pyramid Pooling (ASPP) module consisting of four different DCL. Using different dilatation rates in each atrous convolutional layer, this module probes features at multiple scales of the input feature map. It uses bilinear sampling to reproduce larger image translation. This is not an effective way of upsampling because of its independence from data. Zhi Tian *et al.* on the other hand provide Data-dependent Upsampling (DUpsampling) technique to improve the segmentation performance [25]. They provide a low resolution path to correctly upsample the translation mapping. Instead of computing the loss with a high resolution ground truth Y, it is compressed through linear projection to the size of translation output before bilinear upsampling. They report a mIOU of 88.1% for the PASCAL VOC test set. The disadvantage to this kind of upsampling is its poor generalization. DUpsampling is good for small label space but it is unable to generalize complex or large label space because of its strong data dependency [26]. Huikai Wu *et al.* propose a Joint Pyramid Upsampling (JPU) to address this problem. JPU also confronts the challenge of high computation complexity for DCL [26]. This type of upsampling focusses on learning computationally less expensive and simplified upsampling when it is fed with high resolution guidance and low-resolution target image.

To construct the segmentation map, traditional segmentation networks are largely dependent on high level features representation. The cost of extracting high-level feature representation is reduced spatial resolution. Just considering high-level features is not enough to achieve pixel level translation. Given high-level features, information about the spatial distribution is also needed. This information is mainly available in earlier layers. Skip connections help the model to maintain spatial stability by sharing intermediate information between layers. Given the fact that earlier layers not only possess spatial information but also low-level features, the selection of relevant information is very important. To extract and process the appropriate information Xiangtai Li *et al.* propose Gated Fully Fusion (GFF) scheme [20]. The GFF uses multilevel feature maps to generate enriched features with high resolution. They achieve 82.3% mIOU on Cityscapes dataset. Takikawa *et al.* process the information in regular and shape-based streams to improve the semantic map using boundary information [19]. GCLs are used to extract and fuse the useful information in shape stream. The high-level features from different intermediate layers of regular stream are used to learn the edges of semantic map. The high-level features from regular stream and shape-based stream are finally fused together using ASPP module. They report a mIOU of 80.8% for Cityscapes dataset.

One of the key elements in training DCNN is using enough examples to boost accuracy. Image augmentation like blurring, zooming, and tilting is helpful in improving the results but there can be better ways to achieve this goal. For example, Y. Zhu *et al.* propose a joint adventure of image and label propagation mechanism to scale up training examples [27]. The next pixel location is estimated using motion vectors predicted using vector-based approach [28]. The fine-tuned data is used to train DeepLabv3+ [18] reporting a mIOU of 83.5% on Cityscapes dataset. In [29], a dual ASPP and decoder are used to perform semantic and instance segmentation. There are two major changes introduced in [29] when compared to Deeplabv3+ [18]. One is introduction of an extra low level feature branch for the decoder and second is adding a 5 × 5 depthwise-separable convolution layer after each upsampling stage. With score of 84.2% mIOU, this work concludes that selectivity of low scale features is important and can satisfactorily improve the results.

One of major failures of the state of the art models is around the boundary of object. For instance, in [30], erroneous pixels are concentrated at boundary of the object. The number of mistakes decreases with the increase of distance from the boundary. In [30] a post processing technique is utilized to refine the boundary errors using a direction map. The SegFix [30] technique with HRNet+OCR [31] improves the mIOU by 1% with almost real time efficiency.

Objects at different scales impact the model performance for semantic segmentation. With simple encoder-decoder networks this problem is addressed in [32] and [18], [24], [25] using pyramid pooling and dilated convolution, respectively. This solves the object scaling problem, but introduces gridding artifacts [33], [34]. Empty spaces in dilated filters reduce the strength of the relationship between adjacent pixels. This results in loss of spatial information. Thomas Ziegler *et al.* propose solution to this problem by providing smoothing in DCLs [35]. Before each DCL, interpolation filter is applied on each input channel of the layer. It helps in extracting more local information by introducing role from neighbourhood pixels. These methods rely on the last layer for scaling features which has reduced receptive field. To counter this challenge [36] and [37] leverage intermediate features to understand scaling along with last layer. These techniques use dilation and pooling, which result in patterned scaling. The more aligned scaling is discussed in [31], [38]–[40]. The other way of extracting scaled information is to use a joint network for different scaled images. The information is then fused using different approaches like pooling and attention mechanism [41]–[45]. Recently, A. Tao *et al.* pointed the failure of different segmentation techniques for objects at multiple scales [17]. Generally, for large objects, these techniques perform good for low resolution images and poor for higher ones. In contrast, detection of small objects is good in high resolution images. In [17], attention-based scheme is utilized to blend the information from multi-scale predictions. The results achieved are state-of the art for Cityscape dataset with 85.1%mIOU.

Asad and Bais propose MLC based image labelling mechanism for weeds detection in canola field images [13]. They study different state of the art models to detect weeds in canola fields. They state that SegNet [16] with ResNet-50 [3] backbone performs better for weeds detection reporting a mIOU of 82.88%. This work propose binary segmentation solution with background and weed being the classes of interest. They report a IOU score of 66.48% for weeds. Milioto *et al.* propose a lighter network for weeds detection in sugar beet considering an architecture similar to SegNet [11], [16]. They initially process the 4-Channel input image to extract different important information like indices and gradient. This information is further processed through the network to learn mapping. They report a mIOU of 80.8%. Dilated convolution based networks are also used for weeds and crop segmentation [46]. P. Lottes *et al.* utilize odometry information to select consecutive 5 images in a row. This data is used in sequential module to extract arrangement of crop and weeds. The encoded features from sequential module are used to perform weeds and crop detection. This work reports mean $F1$ score of 92.4%.

We aim to improve the segmentation for weeds detection under a challenging dataset. We provide end to end solution for weeds and crop segmentation including the labelling of images at pixel level. The image labelling is semi-automatic and only useful for agricultural images from same domain. We develop a modified version of U-Net [10] with reduced number of parameters and less memory. At input side, we propose parallel feature extraction mechanism using DCLs. This technique extracts bigger connectivity of low-level features from image using different dilation rate. One can think of dilated filter as an approximation of large kernels but with less parameters. These extracted features are helpful in improving the segmentation using traditional network stream. We convert each convolutional layer in a two-operation layer, point wise and depth wise convolution. This helps in reducing the number of trainable parameters. In next section we discuss the dataset, labelling, detail of the model, and its design rationality.

III. METHODOLOGY

We propose semi-automatic image labelling scheme for agricultural images. We process data through multiple image processing techniques to get a fine pixel level annotation for the three classes, weed, crop, and background. We propose extracting features using padded filter at different scale to get a bigger picture of features. We utilize point wise and depth wise convolution to reduce number of parameters. In the

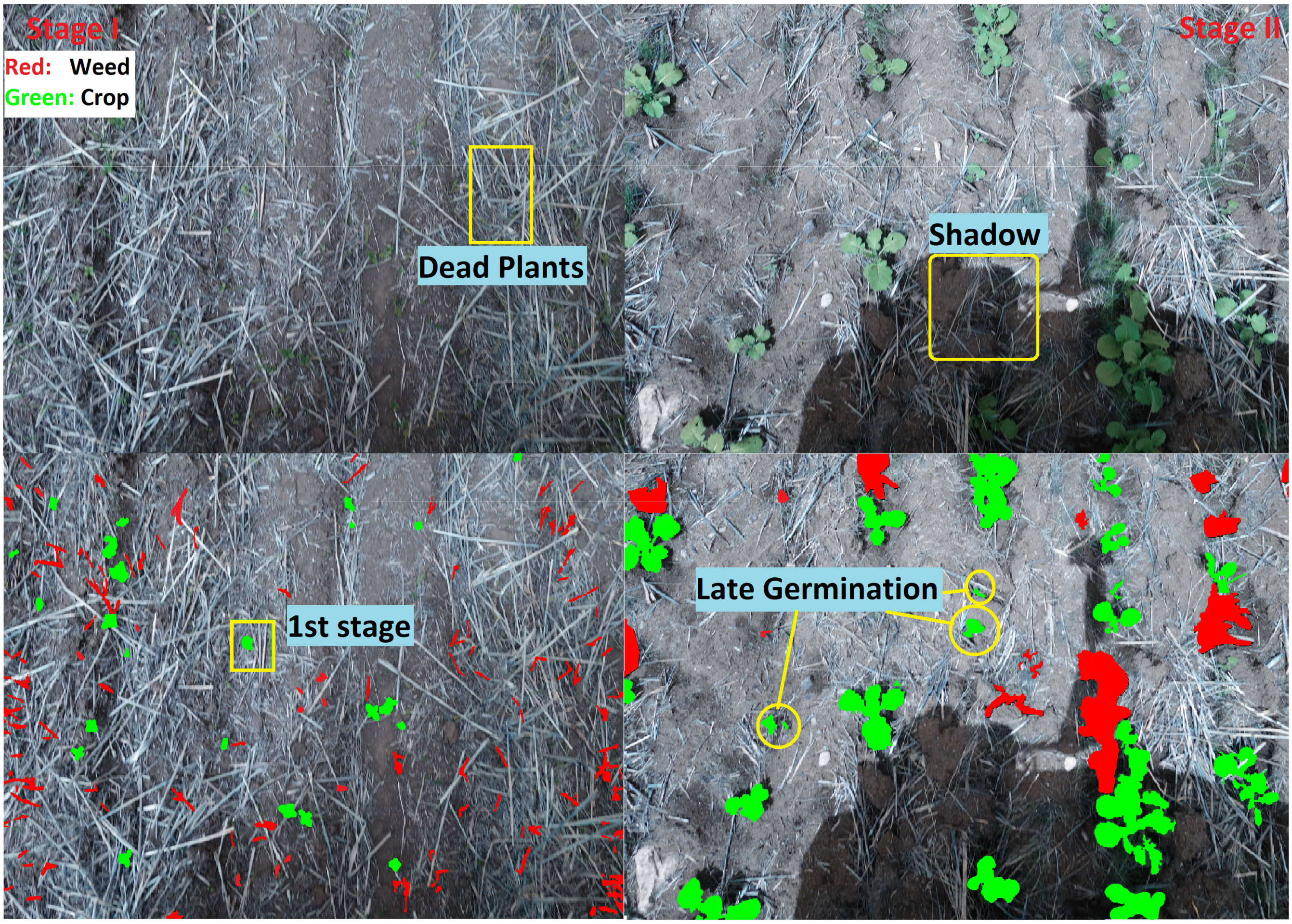


FIGURE 1. Dataset description and visuals: This figure highlights some of the key problem in dataset. The dataset has multiple uncontrolled field environment challenges including but not limited to shadow (yellow rectangle in stage II image), late germinated plants (circled in last section of image) and dead plants. The figure also gives an overview of how difficult this challenge is with changing growth of the plant.

following subsections, we describe the collection of datasets, labelling tool, and the proposed changes in U-Net model.

A. DATASET COLLECTION AND CHALLENGES

Our dataset consists of 1000 images collected from multiple canola fields at two growth stages. The first stage has two unfold leaves and the second stage has five or more leaves. The second stage also has late germination of canola plants. The possible reasons of late germination are rain-fed irrigation, different water levels, and variable rate seeding over the field. The dataset is collected using a simple but high resolution RGB camera mounted on a Quad. The image collection process is designed to address the economic and commercial constraints of data storage and transmission. Varying lighting condition and soil texture, plant and equipment shadows, dead plants, occlusion, blurring, and multiple weed types are some of the challenges in the dataset. We apply multiple augmentation techniques to enhance the dataset. Primarily, we apply blurring, cropping, rotation, horizontal, and vertical flip operation to augment field and corresponding semantic map. Artificially blurring the images helps mitigate the problem associated with real world equipment. Rotation and flipping the images change the distribution of data inside image giving more examples for training.

Figure 1 shows images from two stages and their overlapped labels. The yellow circles and rectangles signify some of the details in the dataset. Particularly, we highlight the shape diversity within a same crop but from two stages. The images having canola plants with four or less leaves are categorized as Stage I. Images containing canola plants with more than four leaves are categorised as Stage II canola. The shadow in Stage II image is one of the challenges which may result in bad performance for manual vegetation extraction. One can also see the overlapping crop and weed plants in the 2nd last row of Stage II image. This is a difficult challenge to address because sometime crop/weed is partially occluded. The missing part is hard to reconstruct unless more information is available. We approximate this problem by assigning the label of top plant. It is difficult to inspect the plants at early stage with naked eye. The yellow circles in Figure 1 show the late germinated plants. We can see that the late germinated plants are not necessarily of the same shape as the canola plants in the Stage I. Given the limited data, the late germinated plants may be minority class and can result in low performance. As the growing plants change shape overtime it is hard to understand the same plant at different stages. We need large number of examples to understand useful mapping.

B. IMAGE LABELLING

Image labelling is the very first task to perform while doing semantic segmentation. It is time consuming and tedious. We may end up labelling a large number of different types of plants due to narrowly timed growth stages of plants. Crop and weeds segmentation for different field type, area, environment, and the stage of the plant require data labelling from scratch. We devise a semi-automatic labelling protocol to quickly and timely address this real-world problem. In the proposed scheme of image labelling task, a field image is passed through a number of image processing techniques to reduce the human effort. In the first step, field image is converted into binary approximation of vegetation and background. The dead plants, stones and soil are separated from the vegetation and classified as background. The vegetation extraction based on indices is not suitable here because of the colour variation and shadows in the images. We tried different indices but using them comes at a cost of losing valuable information. We use MLC technique to classify the image pixels in two classes. MLC is supervised classification technique using Bayes' theorem [47]. As suggested in [13], training samples are taken from the mosaic made from stitching the images. The training samples represent the background and vegetation class. According to Bayes' theorem [15], if we represent class i with c_i and $\boldsymbol{v}$ as a measurement vector we can write probability of class i given the feature vector $\boldsymbol{v}$ as:

$$P\left(c_i|\,\boldsymbol{v}\right) = \frac{P\left(\boldsymbol{v}|c_i\right) \times P(c_i)}{P(\boldsymbol{v})} \tag{1}$$

The probability decides if the feature vector $\boldsymbol{v}$ belongs to class c_i. The goal here is to classify data in two classes, the background and vegetation, making the classification as binary decision. For binary classification we can write [15]:

$$\boldsymbol{v} \in c_i \;\; if \;\; P(c_i\,|\boldsymbol{v}) > P(c_j\,|\boldsymbol{v}) \tag{2}$$

Equation 2 states that if the probability, $P(c_i|\boldsymbol{v})$, is the largest the feature vector $\boldsymbol{v}$ will belong to class c_i. The total probability $P(\boldsymbol{v})$ can be calculated as:

$$P\left(\boldsymbol{v}\right) = \sum_{n=1}^{N} P\left(\boldsymbol{v}|c_i\right) \times P(c_i) \tag{3}$$

This gives us following equation:

$$\boldsymbol{v} \in c_i \;\; if \;\; P\left(\boldsymbol{v}|c_i\right) \times P(c_i) > P\left(\boldsymbol{v}\middle|c_j\right) \times P(c_j) \tag{4}$$

$P\left(\boldsymbol{v}|c_i\right)$ is assumed to have multivariate normal distribution and can be estimated from the training dataset [15]. Multivariate normal distribution for N-dimensional space can be written as:

$$P\left(\boldsymbol{v}|c_i\right) = (2\pi)^{\frac{-N}{2}}\;|\boldsymbol{Y}_i|^{\frac{-1}{2}} e^{-\frac{(v-z_i)^T}{2}}\;\;\boldsymbol{Y}_i^{-1}(v-z_i) \tag{5}$$

Which for binary case simplifies to:

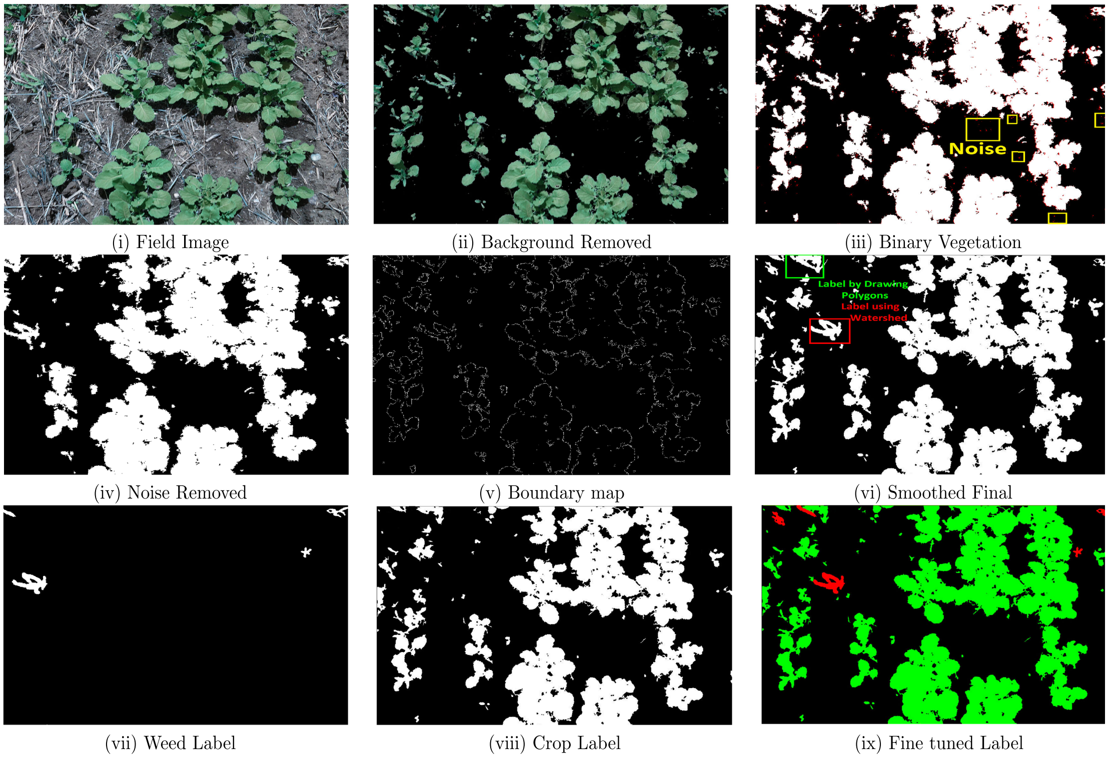
$$P\left(\boldsymbol{v}|c_i\right) = (2\pi)^{\frac{-1}{2}}\;|\boldsymbol{Y}_i|^{\frac{-1}{2}} e^{-\frac{(v-z_i)^T}{2}}\;\;\boldsymbol{Y}_i^{-1}(\boldsymbol{v}-z_i) \tag{6}$$

where z_i and $\boldsymbol{Y}_i$ are mean vector and co-variant matrix, respectively. Once the background is subtracted, images are converted to binary representation with zero as background and white as vegetation. This binary image has noise and may include connected blobs of more than 25 pixels which are not vegetation. Common techniques for noise removal like median or averaging filtering are not very effective. These methods help to remove noise but are not reliable to filter bigger blobs which are neither crop nor weed. To get rid of noise, we first find the connected blobs in the image and calculate their areas. If the area of an 8-connected blob is smaller than 25 pixels then it is removed. Once the image is noise free, the next concern is edges of the blobs. While converting the image into binary the reflection of vegetation effects the boundary of the object. To tackle this problem, we smooth the boundary by image subtraction. We first find the perimeter of all the objects and then subtract it from the noise free image.

Given a background subtracted image we label the easier class by either drawing polygons or using the watershed algorithm. The watershed algorithm [48] is used where there is no occlusion problem. In watershed algorithm the connected pixels can be assigned same value or colour. When one of the class is labelled, it is used to generate the label for other class using image subtraction. Once everything is done we manually correct any missing labels. The process is shown in Figure 2. The red blobs in binary vegetation image represents the noise. The yellow rectangles indicate some of the noisy pixels in the image. This noise is removed by filtering the image based on the area of 8-connected blobs as shown in Figure 2 (iv). The perimeter is extracted (as shown in Figure 2 (v)) from the noise free image. This subtraction results in smoothed image having less reflection around the edges. It is presented in Figure 2 (vi). The green rectangle in Figure 2 (vi) represents the area which we label by drawing polygon. The red rectangle indicates the area where watershed algorithm is more suitable and time efficient. Based on visual inspection of the image, minority class is decided and labelled as shown in Figure 2 (vii). The majority class label is shown in Figure 2 (viii). It is extracted by subtraction of labelled image and noise free vegetation (Figure 2 (vi)). The final fine-tuned label is achieved by combining the crop and the weeds label. It is shown in Figure 2 (ix). This process, on the average takes four to five minutes to label one image. The time depends on the minority class pixels.

C. DILATED U-NET

After completing the image labelling task, the next task is to learn the semantic mapping from the field image.



(i) Field Image (ii) Background Removed (iii) Binary Vegetation

(iv) Noise Removed (v) Boundary map (vi) Smoothed Final

(vii) Weed Label (viii) Crop Label (ix) Fine tuned Label

FIGURE 2. Step by step Semi-automatic Image Labelling for Agriculture Images, the process shows the output of each operation applied to achieve the final label. Vegetation is extracted from the field image using MLC [15] which is then processed through a two-step noise removal process. The minority class is labelled through human annotator, the label for third class is obtained by image subtraction.

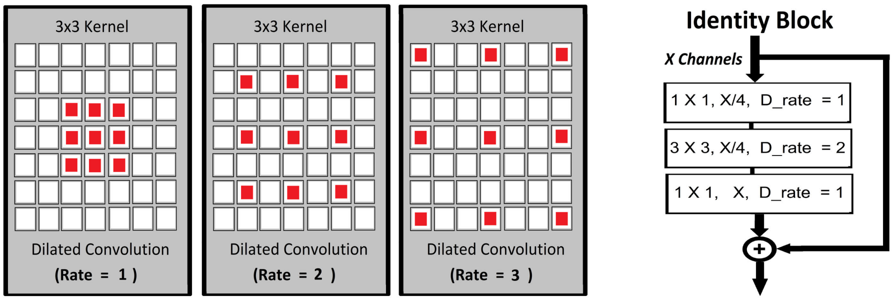
We experiment with basic segmentation networks such as SegNet, U-Net, and Deeplabv3+ with ResNet-50 as backbone. SegNet and U-Net are the well-known architectures for semantic segmentation. Both models use skip connections to share the information from previous layers. The information shared by U-Net is more detailed than that of SegNet. SegNet only shares the pooling indices. These architectures with traditional feature extractors are not reliable for semantic segmentation. One of the reasons of poor performance is losing spatial configuration. While extracting deep and high-level features, we end up having reduced feature space. With reduced feature space it becomes difficult to construct a good object representation as semantic map. Although the skip connections share some of the information to maintain the spatial stability but they are not very helpful.

Another problem of these architectures is their reliance on high level features to generate results. The shape-based feature and high-level information are extracted collectively. This confuses the architecture to perform well. While segmenting the object, boundary of the object is of concern. It makes shape-based features of more interest. To get a bigger picture of the connected blobs, one of the solutions is learning bigger kernel to extract the boundary information. Increasing the filter size adds more parameters to learn and increases the computation cost. The other way to get an approximate bigger picture is to use dilated convolution where padded filters are used to extract the low-level features. If the dilation rate is d, there will be $d-1$ zeros added in the filter. $1D$ dilated convolution for a kernel k is given below [35]:

$$O[i] = \sum_{z=1}^{Z} X[i - d \cdot z]k[z] \tag{7}$$

where O represents output and X is the input vector. A $2D$ visualization of dilated filter with different dilation rates is given in Figure 3(a).

The connectivity of the object and shape information is very important in semantic segmentation. The information about the connectivity of object can be extracted using large kernels, but it is computationally expensive and requires more memory. We assume that if the feature extractor is fed with shape enriched information, it can perform better. Based on our assumption, we propose modified version of U-Net. We add three parallel layers at input side with dilation rate of 2, 6, and 9 to extract features at multiple scales. We use ResNet-50 as feature extractor where in every middle layer of identity block a dilation rate of 2 is used. Figure 3(b) shows the modified identity block in ResNet feature extractor. We use U-Net like skip connections with nearest-neighbour upsampling technique. The feature vectors from skip connection are fused using concatenation layer at different scales of decoder. To reduce the number of parameters, we convert



(a) Dilated Filter representation

(b) Modified Identity Block in ResNet

FIGURE 3. Dilated Filter representation [49] and modification within ResNet feature extractor, in Figure (b) X represents the number of channels in the input and *D_rate* is the dilation rate.

all convolutional layers to be depth wise separable [50]. A depth wise separable layer performs convolution in two steps. In first step, the depth wise spatial convolution is performed. It deals each input channel separately. In second step point wise convolution is performed. The modified network is shown in Figure 4.

Initially, we used cross entropy loss to train our models. This loss is not suitable for semantic segmentation task because the total loss is averaged over all pixels. Given, boundary information, size, and the shape of object as important features, one cannot expect better tuning of the model. Also, if the dataset is unbalanced the task of semantic segmentation becomes more challenging. Although dataset imbalance problem can be addressed using weighted cross entropy, but this loss is still incompatible due to its discrete nature. The discrete nature means it deals with the individual pixel classification but not area overlap measure. DICE loss on the other hand provides a score of how much area of an object is classified correctly. We train our model globally using DICE loss proposed in [51]. With the DICE loss the model adjusts the weights to increase the overlap between prediction and ground truth. The DICE loss can be calculated using the following equation (8):

$$\mathcal{L}(Y_{pred}, Y_{true}) = 1 - \frac{1}{C}\sum_{c=1}^{C} \frac{2\sum_i^N Y_{pred_{i,c}} Y_{true_{i,c}}}{\sum_i^N Y^2_{pred_{i,c}} + \sum_i^N Y^2_{true_{i,c}}} \quad (8)$$

where $Y_{pred} \in [0, 1]$ is predicted probability, the softmax output, and $Y_{true} \in [0, 1]$ is the ground truth. N represents the number of pixels and C denotes the total number of classes. We train the models for 200 epochs and use Adam optimizer with learning rate 0.0001. Initially, we perform our experiment using stepwise learning rate decaying policy. Using this policy may lead to optimization instability because of its discontinuous nature. Then, we change it to polynomial rate decay as it is smoother and leads to better results.

D. ABLATION STUDIES

In this section, we present selection of different architectural components. We conduct several experiments to analyse the design component of the proposed model. We performed all our experiments with ResNet-50 [3] backbone.

We start with simple ResNet-50 based U-Net. We train the model using Categorical Cross Entropy (CCE) loss. Using CCE loss, results in mIOU of 82.4%. Our dataset is highly imbalanced and CCE loss is not suitable as it assigns more weightage to correctly predicted class. Instead of CCE loss we use overlap measure (DICE loss) as penalty. This loss performs well for imbalanced data because it computes loss over misclassified object area. In comparison to CCE, using DICE loss improves the results from 82.4% to 88.73%.

To make the network lighter we convert all the layers to depthwise-separable convolutional layers. This results in less trainable parameters with a compromise on performance reporting 84.36% mIOU. To tackle this challenge, we introduce dilation rate of 2 in middle layer of each of the identity block in ResNet-50. This extracts the alignment related features encoded within the deep feature vector. This improves mIOU by 2%.

With a mIOU of 86.54% we are looking to achieve more robustness in the results. We introduce parallel dilated input layers to extract low level features to help the model to process shape-based features within the same stream. We perform experiments with different number of input layers. While doing experiments, we try to strike balance between performance and model complexity. To accomplish this, we keep the number of filters same i.e., 64 and change the number of input dilated layers. We test 2, 3, and 5 input layers to find a suitable combination. We keep dilation rate at 2, 6, 9, 12, and 15 for each addition of layer respectively. Addition of parallel dilated convolution layers improves the results by more than 2% with minor increment in number of parameters. This is true for all the three combinations of layers. Table 1 gives an overview of performance evaluation using mIOU, number of parameters, Floating Point Operations (FLOP), and memory consumed. We have included the details of modification and the results achieved under specific configuration. We start with a model with memory of 152.7MB and 170.5B of FLOPs. This model has 38.04M parameters that are reduced to 15.22M by converting all the

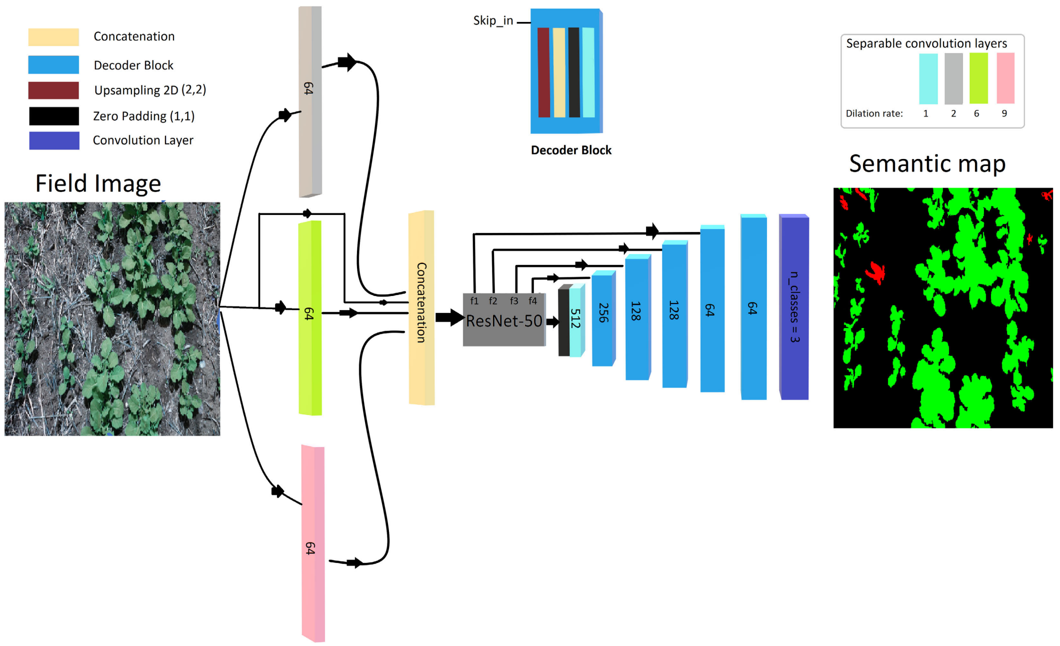


FIGURE 4. Modified U-Net, the skip connections represent the information sharing in the direction of arrows. Decoders has six blocks of upsampling, depth-wise separable, and normalization layers with 512, 256, 128, 128, 64, and 64 filters each. All the layers has filter size 3 × 3 other than the parallel layers where we use 7 × 7, 5 × 5, and 3 × 3 shaped filter. The number of kernels in last layer is equal to number of classes which are in our case 3.

TABLE 1. This table reports the experimental evaluation and design parameters of the model. All the models are based on U-Net (ResNet-50). CCE represents categorical cross entropy, DICE is the overlap measure. DS Convolutional layer represents depth-wise separable convolutional layer. This evaluation represents the detailed designing parameters of the proposed model. The ↓ and ↑ arrows represent the best value trend. The columns having ↓ arrow represent that the minimum is the best. The ↑ arrow represents that highest is best.

Sr.No	Loss Type		DS Conv. Layer	Mod. Identity	Parallel layers			Layer details			Parameters↓	Memory↓	FLOPs↓	mIOU↑	Background↑	Crop↑	Weed↑
	CCE	DICE			2	3	5	Filter size	No. of filters	*D_rate*							
1	✓										38.04M	152.7MB	170.5B	0.8265	0.9714	0.8023	0.7057
2		✓									38.04M	152.7MB	170.5B	0.8873	0.9815	0.8620	0.8183
3		✓	✓								**15.22M**	**61MB**	**49.23B**	0.8436	0.9759	0.7798	0.7753
4		✓	✓	✓							**15.22M**	**61MB**	**49.23B**	0.8654	0.9789	0.8186	0.7987
5		✓	✓	✓	✓			7×7 5×5	64 64	2 6	15.24M	61.5MB	52.75B	0.8868	0.9820	0.8475	**0.8308**
6		✓	✓	✓		✓		7×7 5×5 3×3	64 64 64	2 6 9	15.25M	61.5MB	54.44B	**0.8912**	**0.9823**	**0.8611**	0.8299
7		✓	✓	✓			✓	7×7 5×5 3×3 3×3 3×3	64 64 64 64 64	2 6 9 12 15	15.26M	61.6MB	57.83B	0.8835	0.9817	0.8407	0.8281

layers to depthwise separable convolutional layers. The new size of the model is 61MB. The addition of parallel low feature extraction mechanism uses almost 0.03M parameters making the total parameters 15.25M and memory 61.5MB. Provided the less trainable parameters the FLOPs drops from 170.5B to 54.44B which is more than 3× drop. Although using two parallel layers has less parameters and FLOPs compared to three and five but it has less performance score. With only two parallel layers, model reports an overall mIOU score of 88.68%. Adding three parallel layers instead of two, increases the parameters and FLOPs by 0.01 M and 1.69 B respectively. The results are promising for three layers with an improvement of 0.44%. Adding five layers instead of three or two does not improve performance. We select the best trade off between all the limits and choose the model with three dilated parallel layers. The architecture with three parallel layers achieves 2.58% overall improvement. The chosen model shows nearly equal performance compared to best performing heavy weight model.

IV. RESULT AND DISCUSSION

We perform extensive experimentation to evaluate different models. We use 75% of the data for training the model and 15% for testing, and the rest of the data is used to validate the

TABLE 2. **Comparison of different models with our model is briefly described in above table. Based on the results above, our model is more confident in detecting weeds and background with a score of 92.4% for weeds and 99.45% for background. In overall picture, HRNet_Mscale performs better with mIOU of 90.34%.**

Model	mean IOU↑	Overlap Measure (IOU)			mean AP↑	Average Precision (AP)		
		Background	Crop	Weeds		Background	Crop	Weeds
U-Net simple [10]	0.8971	**0.9834**	0.8665	0.8432	0.9165	0.9912	0.8923	0.8661
U-Net (ResNet-50) [10]	0.8873	0.9815	0.8620	0.8183	0.9353	0.9916	0.9235	0.8908
SegNet (ResNet-50) [16]	0.8778	0.9804	0.8354	0.8176	0.9232	0.9911	0.8937	0.8847
DeepLabv3+ (ResNet-50) [18]	0.8958	0.9829	0.8685	0.8359	0.9417	0.9930	0.9308	0.9014
HRNet_Mscale [17]	**0.9034**	0.9633	**0.8932**	**0.8536**	0.9476	0.9806	**0.9521**	0.9112
Modified U-Net (Ours)	0.8912	0.9823	0.8611	0.8299	**0.9533**	**0.9945**	0.9410	**0.9245**

model. Keeping in view the memory constraints, we split the high-resolution images in four tiles of size 800 × 512 each. We use this data to train the model and save the model weights with best validation set accuracy. Once the model training is done, we collect our quantitative performance measure on test data. Accuracy is not a good measure for semantic segmentation. Although it is classification task but given the imbalanced data, even an accuracy of more than 90% does not mean anything as most of the pixels belongs to background class. For this reason, we use IOU score and Average Precision (AP) to compare the performance of the models on test data. AP gives a measure of confidence of model under different thresholds. Whereas, IOU score measures the overlap between the ground truth and prediction. It can be calculated by Equation 9:

$$IOU = \frac{Y_{true} \bigcap Y_{pred}}{Y_{true} \bigcup Y_{pred}} \tag{9}$$

We report the individual and mean IOU score for weeds and crop detection in Table 2. HRNet_Mscale [17] has the best numbers with mIOU of 90.34% with 89.32%, 85.36%, and 96.33% for crop, weed, and background, respectively. Our proposed modifications in U-Net model not only achieve comparable performance but also reduces the number of trainable parameters. We have included AP in Table 2 for individual class. In AP, the proposed model show overall higher confidence by 0.5% comparing with the second-best benchmark model. The important thing is to note the AP for weeds classification, for the proposed model it is higher by 1.33% compared to state of art indicating our model is more confident in weeds detection.

HRNet_Mscale [17] with mIOU of 90.34% shows exceptional results. For IOU, it outperforms in crop and weeds detection by 2.47% and 1.04% respectively compared to second best. In contrast, the overall AP is higher for modified-U-Net. Our model is more certain about the positive detection given the mAP of 95.33%. The proposed model lags in AP for crop class by a score of almost 1% in comparison with HRNet_Mscale. Although HRNet_Mscale shows good performance for overlap measure but it is less certain about the class of specific pixel. It is also not practical for our application which requires a light model with a given compromise on accuracy. Our model shows nearly equal performance compared to other state of the art models having more parameters. We miss some of the late germinated plants in the labelled data. Majority of the examined models not only understand the labelled data but also help in improving the ground truths by pointing out the mistakes or missed labels. The trained model can also be used to ease the labelling process. The model trained on small data can help reduce the effort needed to label the minority class pixels. With no or very little fine-tuning, one can label the images efficiently.

TABLE 3. **This table represents comparison of model performance with different memory and computation power constraints. Our model has only 15*M* trainable parameters and barely occupies space on a remote device. It is also computationally efficient so provides time efficient response.**

Model	mIOU↑	Parameters↓	FLOPs↓	Memory↓
U-Net simple [10]	0.8971	31.04 M	683.21 B	124 MB
U-Net (ResNet-50) [10]	0.8873	38.04 M	170.46 B	152.7 MB
SegNet (ResNet-50) [16]	0.8778	34.65 M	121.38 B	139 MB
DeepLabv3+ (ResNet-50) [18]	0.8958	24.81 M	132.58 B	96.3 MB
HRNet_Mscale [17]	**0.9034**	72.1M	165.73 B	578.4 MB
Modified U-Net (Ours)	0.8912	**15.25 M**	**54.44 B**	**61.5 MB**

Table 3 shows comparison of the models based on their evaluation score (mIOU) and computational complexity. The proposed model only takes 15*M* parameters and perform nearly equal to best performing model. The proposed model reports 89.12% mIOU with 86.11%, 86.11%, and 98.23% for crop, weed, and background, respectively. There is a compromise of 1% overlap score compared to best performing model with an achievement of 56.85*M* reduction in parameter. The proposed model only require 61.5*MB* storage and perform 54.44B multiply and add operation given one image. The FLOPs for best performing model are 2.3× more than the proposed model making it unfavourable choice for a near real time response. The memory requirement for model storage and deployment is itself significantly low almost by a factor of 10 compared to the best performing model.

The qualitative results are shown in Figure 5 and Figure 6. The green represents crop and red colour indicates weeds. As visible from the images in Figure 5, the soil texture and crop size are varying. The figure reports the results based

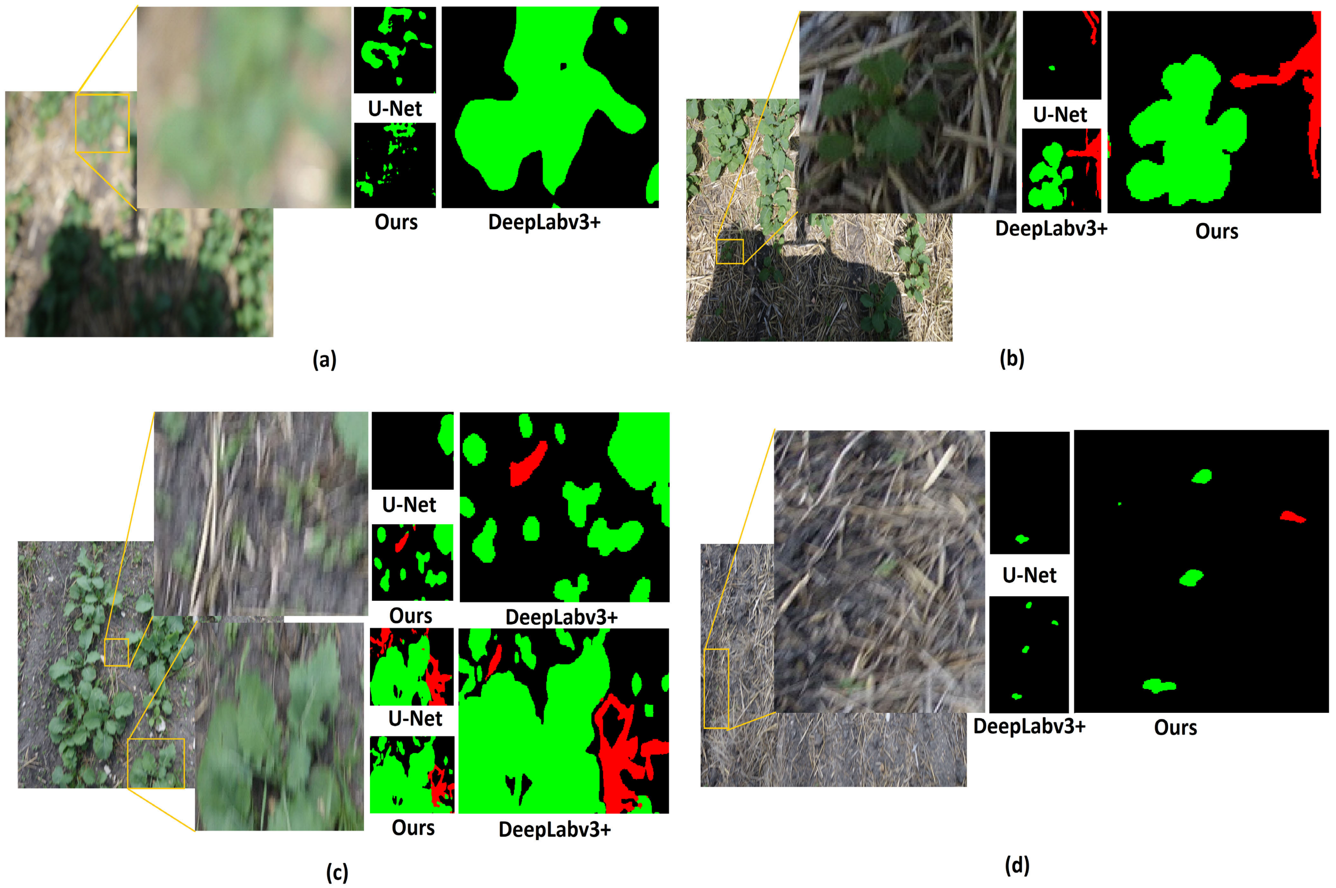


FIGURE 5. Qualitative results of three top performing models on different challenging images in the dataset. The figure shows results on blurry image, image with shadow, multistage, and early-stage crop. In Fig. (a) DeepLabv3+ shows best performance for blurry images. Fig. (b) reports performance on images with shadow, the propose model works better compared to Deeplabv3+ and U-Net. Part (c) shows visual results for images with multistage plants, U-Net does not detect late germinated plants whereas Deeplabv3+ shows best performance. In Fig. (d) we report results on different soil texture, the proposed model effectively captures this variance in soil and outperforms DeepLabv3+.

on complex scenarios related to technical and environmental constraints. As discussed earlier the image acquisition mechanism can result in images with varying light, blurring of the image, and shadow. These limitations are visualized in Figure 5(a) & (b).

In blurred images, the difficulty is to exactly estimate the object's class. Our proposed model does not perform well for these images. The likely reason is the information loss in extracting deep features. DeepLabv3+ [18] shows better performance on blurry images. An example is shown in Figure 5 (a). Shadows in the images also make the segmentation task harder unless enough good examples are provided. The U-Net [10], and SegNet [16] with ResNet-50 backbone miss important information under the shade. Our proposed model has better performance for images with shadow. Also, it is evident from Figure 5 (b) that the U-Net mislabel the object under shade. It is classified as background. On the other hand, DeepLabv3+ and our modified model correctly identify the presence of an object and its class that is the crop in this case.

Shape of crop and weeds is changing with growth making it a challenging task for the model to segment plant at multiple stages. In Figure 5(c) we present example image with crop and weeds at different growth stages. U-Net [10], fails to identify crop labels at early stage. In images with multi-stage crop, U-Net [10], classify early-stage crop plants either as background or weed, it gives poor comparison in term of generalization. DeepLabv3+ [18] achieve better results in detecting the weeds and crop at multiple stages (late germinated plants) as shown in Figure 5(c).

Our model (modified U-Net) outperforms the remaining models for changing background texture as shown in Figure 5(d). The soil texture primarily varies due to different amount of water in the soil. Variations in background reduce the model performance. As discussed earlier, dataset contains two stages of canola crop. Results from both stages are visualized in Figure 6. The second stage has late germinated plants in which some of them are barely recognizable. If we compare both stages, we can see that the first stage has some resemblance with late germinated plants. This resemblance

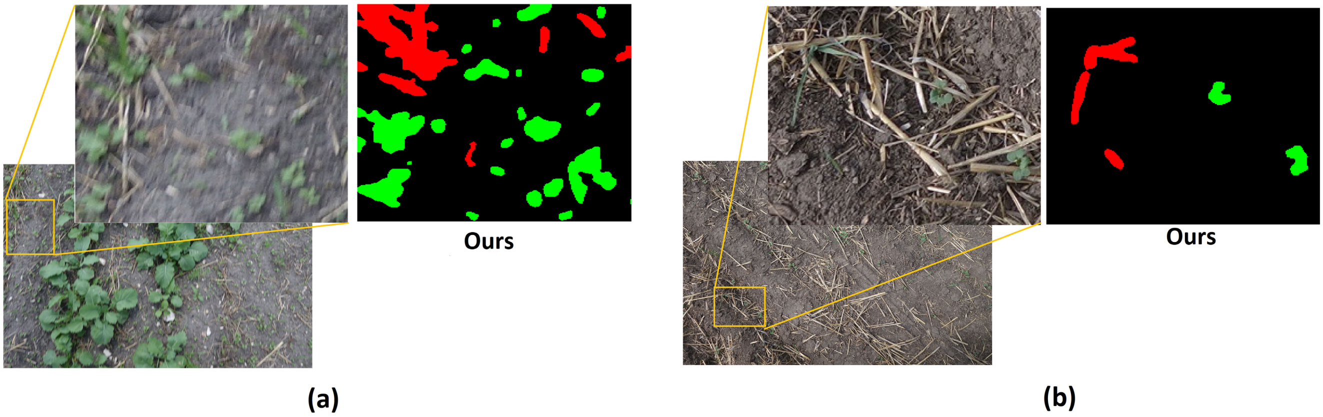


FIGURE 6. The late germinated crop and early-stage crop detection: The image in Fig. (a) shows prediction of late germinated plants available with in later stage crop. Fig. (b) shows performance of the proposed model on early-stage crop. Our model shows better performance when only one stage is available in the given image.

is not very prominent as the late germinated plants are from multiple growth stages.

Looking at overall picture and visual examples it can be concluded that DeepLabv3+ [18] outperforms other models. U-Net [10] although provides second best mIOU score but it is unable to detect the mislabelling in the test data. The crop IOU for HRNet_Mscale [17] signifies that performance of the model for crop is by far the best among the compared models. In the end, given weeds detection as an essential factor for variable rate herbicide application, our proposed model provides a better trade-off between generalization, memory, quantitative, and qualitative results.

V. CONCLUSION

In this paper, we proposed a semi-automatic image labelling mechanism at pixel level. This scheme is suitable for weeds and crop discrimination in agriculture images. Field images are classified in background and vegetation which are then fine-tuned to remove noise. We label minority class by drawing polygons and use image subtraction to get the other class labelled. To achieve an easily deployable, and memory and response efficient model, we modify classic U-Net model. The modification achieves comparable performance score. We extract features from the given images using dilated convolution with three different dilation rates. We report a mIOU of 89.12% for test data. Overall, in comparison with the best score, there is a decrease of 1.2% in mIOU score. Our model uses 57M less parameters than HRNet_Mscale [17].

There is still need of improvement in accelerating labelling mechanism. We spend 4-5 minutes per image for labelling which can become problematic task for large number of images. We are exploring other ways of easing the labelling process as proposed in [52]. This work needs efforts on preparing the dataset. Other possibilities of extending this work are, looking into a multi class segmentation task where we assign class for each of the weed. Again, it requires tons of images labelled at pixel level which is a time consuming task.

REFERENCES

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.

[2] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*. [Online]. Available: http://arxiv.org/abs/1409.1556

[3] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[4] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2818–2826.

[5] A. Ess, T. Mueller, H. Grabner, and L. V. Gool, "Segmentation-based urban traffic scene understanding," in *Proc. Brit. Mach. Vis. Conf. (BMVC)*, vol. 1, Sep. 2009, p. 2.

[6] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 3354–3361.

[7] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele, "The cityscapes dataset for semantic urban scene understanding," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3213–3223.

[8] M. Teichmann, M. Weber, M. Zollner, R. Cipolla, and R. Urtasun, "MultiNet: Real-time joint semantic reasoning for autonomous driving," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 1013–1020.

[9] G. Litjens, T. Kooi, B. E. Bejnordi, A. A. A. Setio, F. Ciompi, M. Ghafoorian, J. A. W. M. van der Laak, B. van Ginneken, and C. I. Sánchez, "A survey on deep learning in medical image analysis," *Med. Image Anal.*, vol. 42, pp. 60–88, Dec. 2017.

[10] 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, Oct. 2015, pp. 234–241.

[11] A. Milioto, P. Lottes, and C. Stachniss, "Real-time semantic segmentation of crop and weed for precision agriculture robots leveraging background knowledge in CNNs," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2018, pp. 2229–2235.

[12] I. Sa, M. Popović, R. Khanna, Z. Chen, P. Lottes, F. Liebisch, J. Nieto, C. Stachniss, A. Walter, and R. Siegwart, "WeedMap: A large-scale semantic weed mapping framework using aerial multispectral imaging and deep neural network for precision farming," *Remote Sens.*, vol. 10, no. 9, p. 1423, Sep. 2018.

[13] M. H. Asad and A. Bais, "Weed detection in Canola fields using maximum likelihood classification and deep convolutional neural network," *Inf. Process. Agricult.*, vol. 7, no. 4, pp. 535–545, Dec. 2020.

[14] N. Chebrolu, P. Lottes, A. Schaefer, W. Winterhalter, W. Burgard, and C. Stachniss, "Agricultural robot dataset for plant classification, localization and mapping on sugar beet fields," *Int. J. Robot. Res.*, vol. 36, no. 10, pp. 1045–1052, Sep. 2017.

[15] P. S. Sisodia, V. Tiwari, and A. Kumar, "Analysis of supervised maximum likelihood classification for remote sensing image," in *Proc. Int. Conf. Recent Adv. Innov. Eng. (ICRAIE)*, May 2014, pp. 1–4.

[16] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[17] A. Tao, K. Sapra, and B. Catanzaro, "Hierarchical multi-scale attention for semantic segmentation," 2020, *arXiv:2005.10821*. [Online]. Available: http://arxiv.org/abs/2005.10821

[18] 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)*, Sep. 2018, pp. 801–818.

[19] T. Takikawa, D. Acuna, V. Jampani, and S. Fidler, "Gated-SCNN: Gated shape CNNs for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 5229–5238.

[20] X. Li, H. Zhao, L. Han, Y. Tong, S. Tan, and K. Yang, "Gated fully fusion for semantic segmentation," in *Proc. AAAI Conf. Artif. Intell.*, Apr. 2020, vol. 34, no. 7, pp. 11418–11425.

[21] C. Liu, L.-C. Chen, F. Schroff, H. Adam, W. Hua, A. L. Yuille, and L. Fei-Fei, "Auto-DeepLab: Hierarchical neural architecture search for semantic image segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 82–92.

[22] R. Shin, C. Packer, and D. Song, "Differentiable neural network architecture search," in *Proc. 6th Int. Conf. Learn. Represent.*, 2018. [Online]. Available: https://openreview.net/forum?id=BJ-MRKkwG

[23] H. Liu, K. Simonyan, and Y. Yang, "DARTS: Differentiable architecture search," 2018, *arXiv:1806.09055*. [Online]. Available: http://arxiv.org/abs/1806.09055

[24] 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. 2018.

[25] Z. Tian, T. He, C. Shen, and Y. Yan, "Decoders matter for semantic segmentation: Data-dependent decoding enables flexible feature aggregation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3126–3135.

[26] H. Wu, J. Zhang, K. Huang, K. Liang, and Y. Yu, "FastFCN: Rethinking dilated convolution in the backbone for semantic segmentation," 2019, *arXiv:1903.11816*. [Online]. Available: http://arxiv.org/abs/1903.11816

[27] Y. Zhu, K. Sapra, F. A. Reda, K. J. Shih, S. Newsam, A. Tao, and B. Catanzaro, "Improving semantic segmentation via video propagation and label relaxation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 8856–8865.

[28] F. A. Reda, G. Liu, K. J. Shih, R. Kirby, J. Barker, D. Tarjan, A. Tao, and B. Catanzaro, "SDC-Net: Video prediction using spatially-displaced convolution," in *Proc. ECCV*, Sep. 2018, pp. 718–733.

[29] B. Cheng, M. D. Collins, Y. Zhu, T. Liu, T. S. Huang, H. Adam, and L.-C. Chen, "Panoptic-DeepLab: A simple, strong, and fast baseline for bottom-up panoptic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 12475–12485.

[30] Y. Yuan, J. Xie, X. Chen, and J. Wang, "Segfix: Model-agnostic boundary refinement for segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Cham, Switzerland: Springer, Aug. 2020, pp. 489–506.

[31] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," 2019, *arXiv:1909.11065*. [Online]. Available: http://arxiv.org/abs/1909.11065

[32] 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.

[33] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell, "Understanding convolution for semantic segmentation," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Mar. 2018, pp. 1451–1460.

[34] Z. Wang and S. Ji, "Smoothed dilated convolutions for improved dense prediction," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 2486–2495.

[35] T. Ziegler, M. Fritsche, L. Kuhn, and K. Donhauser, "Efficient smoothing of dilated convolutions for image segmentation," 2019, *arXiv:1903.07992*. [Online]. Available: http://arxiv.org/abs/1903.07992

[36] D. Lin, D. Shen, S. Shen, Y. Ji, D. Lischinski, D. Cohen-Or, and H. Huang, "ZigZagNet: Fusing top-down and bottom-up context for object segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 7490–7499.

[37] J. Fu, J. Liu, Y. Wang, Y. Li, Y. Bao, J. Tang, and H. Lu, "Adaptive context network for scene parsing," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6748–6757.

[38] Y. Li and A. Gupta, "Beyond grids: Learning graph representations for visual recognition," in *Proc. NIPS*, Dec. 2018, pp. 9245–9255.

[39] K. Yue, M. Sun, Y. Yuan, F. Zhou, E. Ding, and F. Xu, "Compact generalized non-local network," 2018, *arXiv:1810.13125*. [Online]. Available: http://arxiv.org/abs/1810.13125

[40] F. Zhang, Y. Chen, Z. Li, Z. Hong, J. Liu, F. Ma, J. Han, and E. Ding, "ACFNet: Attentional class feature network for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 6798–6807.

[41] 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

[42] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3640–3649.

[43] S. Yang and G. Peng, "Attention to refine through multi scales for semantic segmentation," in *Proc. Pacific Rim Conf. Multimedia*. Cham, Switzerland: Springer, Sep. 2018, pp. 232–241.

[44] 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.

[45] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "CCNet: Criss-cross attention for semantic segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 603–612.

[46] P. Lottes, J. Behley, A. Milioto, and C. Stachniss, "Fully convolutional networks with sequential information for robust crop and weed detection in precision farming," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 2870–2877, Oct. 2018.

[47] A. Ahmad and S. Quegan, "Analysis of maximum likelihood classification on multispectral data," *Appl. Math. Sci.*, vol. 6, no. 129, pp. 6425–6436, 2012.

[48] A. Bieniek and A. Moga, "An efficient watershed algorithm based on connected components," *Pattern Recognit.*, vol. 33, no. 6, pp. 907–916, Jun. 2000.

[49] C. S. Perone, E. Calabrese, and J. Cohen-Adad, "Spinal cord gray matter segmentation using deep dilated convolutions," *Sci. Rep.*, vol. 8, no. 1, pp. 1–13, Apr. 2018.

[50] L. Kaiser, A. N. Gomez, and F. Chollet, "Depthwise separable convolutions for neural machine translation," 2017, *arXiv:1706.03059*. [Online]. Available: http://arxiv.org/abs/1706.03059

[51] C. H. Sudre, W. Li, T. Vercauteren, S. Ourselin, and M. J. Cardoso, "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations," in *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support*. Cham, Switzerland: Springer, Sep. 2017, pp. 240–248.

[52] D. Acuna, H. Ling, A. Kar, and S. Fidler, "Efficient interactive annotation of segmentation datasets with Polygon-RNN++," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 859–868.

HAFIZ SAMI ULLAH received the B.Sc. degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 2018. He is currently pursuing the M.A.Sc. degree in electronic systems engineering from the University of Regina, Canada. He is also working on precision farming applications. His research interests include anomaly detection, video processing, machine learning, computer vision, and deep learning.

MUHAMMAD HAMZA ASAD received the B.Sc. degree in electrical engineering from the University of Engineering and Technology, Lahore, Pakistan, in 2009, and the M.A.Sc. degree in electronic systems engineering from the University of Regina, SK, Canada, in 2019, where he is currently pursuing the Ph.D. degree in electronic systems engineering. He is also working on site specific biotic and abiotic stress management. His research interests include signal processing, machine learning, computer vision, artificial intelligence algorithms, precision agriculture, and predictive maintenance.

ABDUL BAIS (Senior Member, IEEE) received the M.Sc. degree in electrical engineering from the University of Engineering and Technology, Peshawar, Pakistan, in 2003, and the Ph.D. degree in electrical engineering and information technology from the Vienna University of Technology, Vienna, Austria, in 2007.

From 2010 to 2013, he was a Postdoctoral Fellow with the Faculty of Engineering and Applied Science, University of Regina, SK, Canada, where he has been an Assistant Professor with the Electronic Systems Engineering Program, since 2015. He is currently a Certified Instructor with the NVIDIA Deep Learning Institute (Fundamentals of deep learning for computer vision and fundamentals of deep learning for multiple data types). He is also a Licensed Professional Engineer in SK, Canada. His research interests include real-time data stream mining, deep learning, signal processing, image processing, and computer vision.