Plant Growth Prediction using Convolutional LSTM

Shunsuke Sakurai[1], Hideaki Uchiyama[2], Atsushi Shimada[1] and Rin-ichiro Taniguchi[1]
[1]*Graduate School and Faculty of Information Science and Electrical Engineering, Kyushu University,*
744 Motooka Nishi-ku, Fukuoka, Japan
[2]*Library, Kyushu University, 744 Motooka Nishi-ku, Fukuoka, Japan*

Keywords: Deep Learning, Plant Growth, Convolutional LSTM, Frame Prediction.

Abstract: This paper presents a method for predicting plant growth in future images from past images, as a new phenotyping technology. This is achieved by modeling the representation of plant growth based on neural network. In order to learn the long-term dependencies in plant growth from the images, we propose to employ a Convolutional LSTM based framework. Especially, We apply an encoder-decoder model inspired by a framework on future frame prediction to model the representation of plant growth effectively. In addition, we propose two additional loss terms to put the constraints on shape changes of leaves between consecutive images. In the evaluation, we demonstrated the effectiveness of the proposed loss functions through the comparisons using labeled plant growth images.

1 INTRODUCTION

To improve plant harvesting in commercial agriculture, it is important to understand how the environmental conditions affect plant growth. Plant phenotyping is a research issue to deal with this problem in the field of agriculture (Walter et al., 2015). Basically, a plant phenotype, which corresponds to the biochemical and physical appearance characteristics, is affected by the interactions between genetic properties and environmental conditions. Since it differs according to plant species, it is important to measure the relationship between phenotypes and environmental conditions for each plant species. To solve this problem, the development of plant phenotyping systems for various plant species has been conducted for years.

Image based automatic plant phenotyping systems have been developed owing to the advent of various types of low-cost cameras with the advance of computer vision technologies. The advantage of image based approaches has the following two aspects: they are in a non-destructive way, and also allow to continually observe plant phenotype in high-throughput (Li et al., 2014). Traditionally, the simple structures of a plant such as height, center of mass, convex hull have been measured from the images. The recent advance of machine learning techniques such as deep learning allows pixel-by-pixel plant region segmentation in the images (Sakurai et al., 2018b), and plant age estimation from images (Ubbens and Stavness, 2017). In the workshop on computer vision problems in plant phenotyping (CVPPP) workshops, which started since 2014 and are organized by International Plant Phenotyping Network(IPPN) [1], leaf segmentation and counting challenges have been organized to further activate this field. Since this field is absolutely not matured yet, only a small part of phenotypes has been clarified in the literature. Therefore, it is important to investigate further phenotyping technologies for clarifying the fundamentals in agriculture by computer vision techniques.

As a new type of image based phenotyping technologies, we propose a method for predicting plant growth from images. More precisely, the goal is to predict the shape of leaves in the future images at the pixel level from the past images rather than only predicting the size of leaves. To achieve this goal, it is necessary to model how a plant shape geometrically changes as the time passes. In the field of computer vision, a deep learning based method for learning video representation was proposed to predict future images from past images in a video (Srivastava et al., 2015). Therefore, we tackle plant growth prediction by following the video representation lear-

[1] https://www.plant-phenotyping.org/

Sakurai, S., Uchiyama, H., Shimada, A. and Taniguchi, R.
Plant Growth Prediction using Convolutional LSTM.
DOI: 10.5220/0007404901050113
In *Proceedings of the 14th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2019)*, pages 105-113
ISBN: 978-989-758-354-4

ning. In particular, we propose to employ an encoder-decoder architecture of Convolutional LSTM. The primary task of the encoder is to generate the representation of the plant growth from images. This representation is important for the prediction tasks because the plant growth shows large diversity depending on the surrounding environment. In order to model the diversity more appropriately, we propose two additional loss functions for the neural network to put the constraints on the plant growth model. In the evaluation, we demonstrate the effectiveness of our proposed loss functions through the comparisons among four different settings using labeled plant growth images in the KOMATSUNA dataset (Uchiyama et al., 2017), which contains the images of a Japanese leaf vegetable.

2 RELATED WORK

There are several method utilizing chlorophyll fluorescence imaging for modeling plant growth (Barbagallo et al., 2003; Moriyuki and Fukuda, 2016). The effectiveness of the chlorophyll fluorescence imaging to identify the perturbations of leaf metabolism was demonstrated (Barbagallo et al., 2003). Several features of seedlings that included circadian rhythm were extracted based on chlorophyll fluorescence imaging (Moriyuki and Fukuda, 2016). They explored seedling diagnosis by applying a machine learning technique to the features. A neural network was also employed to predict plant growth (Zaidi et al., 1999). They constructed a model of relationship between plant growth and its characteristic.

Next, computer vision techniques related to our method are summarized. Recently, deep learning based methods have achieved state-of-the-art performances in various computer vision tasks. Among them, our plant growth prediction task can be related to both visual future prediction and generative model of images as follows. For the visual future prediction, Recurrent Neural Network (RNN) was used to predict future frames by learning the video representation inspired by a language modeling method (Ranzato et al., 2014). They proposed to quantize small patches into a dictionary, and to use a language modeling method. However, it is difficult to learn the long-term dependencies with RNN. Therefore, the architecture of Long-Short Term Memory (LSTM) (Hochreiter and Schmidhuber, 1997) that is an improved RNN to learn long-term dependencies of video frames was also used (Srivastava et al., 2015). Especially, they used a LSTM encoder-decoder architecture to effectively learn the video representation. The LSTM is extended to Convolutional LSTM for effectively modeling spatial-temporal relationship of images (Xingjian et al., 2015).

For the generative model to synthesize an image with respect to a specific target, Generative Adversarial Network (GAN) can generate highly-sharp and detailed images (Goodfellow et al., 2014). They proposed an adversarial loss that minimized JensenShannon (JS) divergence between input data distribution and generated data distribution. A super-resolution method with very deep network was proposed (Ledig et al., 2017). They showed that adding the adversarial loss allowed a generated image to avoid blur from a Mean Squared Error (MSE) loss. The adversarial loss was applied to semantic segmentation (Luc et al., 2016) . They presented the effectiveness of the adversarial loss even if the task was classification such as semantic segmentation without the MSE.

Several methods on the prediction tasks employed the adversarial loss to improve the prediction performance. A multi-scale network was proposed to predict a future frame (Mathieu et al., 2016). They used the adversarial loss to obtain the prediction with the sharpness of the images. A method for separating the images in terms of the motions and the contents was proposed (Villegas et al., 2017) . They employed a LSTM encoder-decoder architecture, and showed the effectiveness of suppressing the blurring effect with such networks.

In this paper, we propose plant growth prediction network inspired by a framework on the future frame prediction. Basically, both plant growth prediction and future frame prediction share the same research topic in terms of the prediction. However, there are differences from the following two aspects. First, the plant growth prediction is an instance-wise predicting task such as growth of each leaf in a whole plant. Second, the plant growth has very long-term such as over ten days. Therefore, we investigate how the plant growth prediction benefits from the future frame prediction.

3 METHOD

In this section, we describe the detail of our proposed network for plant growth prediction tasks. Figure 1 illustrates the overview of our network architecture. Our method is based on a frame prediction network for the videos in terms of the prediction of future images as output from several past images as inputs. As our technical contributions, several aspects of the network are improved, and the difference are summarized as follows.

The first difference is to use both RGB images and labeled leaf images as inputs, as provided in (Uchiyama et al., 2017). The labeled leaf images are greatly helpful to extract plant traits, and also more useful than RGB images in terms of plant phenotyping. Because of this, we decided to utilize labeled leaf images in our network. Since there are features obtained from only RGB images such as color or curvature, our proposed network takes both RGB and labeled images as inputs and outputs.

The second difference is that the interval between frames is longer than other frame prediction tasks. Normally, the time interval between typical video frames is less than a few milliseconds. However, the interval between plant growth images is longer. For example, the interval is several hours between plant growth images, and the leaf movement is relatively large. Since the assumption that difference is infinitesimal is not satisfied in such a situation, methods based on optical flow (Liu et al., 2018) are not suitable. Instead of using optical flow, we propose a difference loss as a change constraint over frames. In addition, we propose a centroid loss as a constraint on the leaf movement directly. The details are explained in Section 3.3 and Section 3.4.

3.1 Encoder-decoder Network

Srivastava et al. predicted future frames with a LSTM encoder-decoder framework by learning video representation (Srivastava et al., 2015). The encoder-decoder network allows to efficiently learn the representation of inputs domain (Sutskever et al., 2014). The encoder LSTM constructs the representation of the input from a sequence of frames, whereas the decoder LSTM reserves the representation and predicts future frames.

3.1.1 Encoder

The overview of the encoder is illustrated at the left side of Figure 2. Before the encoding process, each input image is fed into convolutional layers. The encoder constructs the representation of inputs over time-steps. An input image runs through the encoder in each-time step one by one. In the final step, the encoder learns the representation of the past images in its cell. Then, this representation is fed to the decoder ConvLSTM cell, and the decoder predicts future images based on this representation. We do not utilize any outputs of the encoder to update network weights explicitly. Updating the encoder relies on backpropagation from the decoder.

3.1.2 Decoder

The overview of the decoder is illustrated at the right side of Figure 2. The decoder LSTM predicts time-series future images. The first frame prediction is generated based on the representation constructed by the encoder. In the first step, the last frame of inputs fed into the encoder is used as a first input frame to the decoder. Other than the first step, zeros are used as inputs of the decoder because the decoder predicts the next step using a recurred previous step output as an input of the next step. This recurred output does not represent outputs after the transposed convolution layer, but represent hidden layer outputs of ConvLSTM. In each time-step, the decoder predicts a next frame. Finally, convolutional layers and bilinear upsampling receives all predicted frames one by one. After this upsampling, we finally acquire the probability map of leaf labels at each pixel.

3.2 Using RGB and Labeled Leaf Images

In addition to RGB images, labeled leaf images are used as both inputs and outputs. As illustrated in Figure 1, a sequence of prediction Y is obtained as outputs of network from a sequence of input data X. In this process, we use multiple frames for both inputs and outputs. The number of the input frames and the number of the output frames can be different. In the following, X_t denotes t-th frame of input data. X^{rgb} denotes RGB images and X^{label} denotes labeled leaf images. The same rule applies to Y.

In our network, RGB images and labeled leaf images are concatenated along channel axis after feature extraction by respective convolutional layers. This is because features of RGB images and those of labeled leaf images should be different. It should be noted that this type of input concatenation is not always best.

The loss functions $\mathcal{L}$ to optimize similarity between prediction Y_t and corresponding ground truth X_t are defined by MSE and multi class cross entropy $H(\cdot)$ as follows.

$$\mathcal{L}_{rgb}(X_t, Y_t) = MSE(X_t^{rgb}, Y_t^{rgb}) \quad (1)$$

$$\mathcal{L}_{label}(X_t, Y_t) = H(X_t^{label}, Y_t^{label}) \quad (2)$$

3.3 Difference Loss

The loss functions described in Section3.2 deal with only similarity in a single frame. In other words, they do not consider sequential constraints. As an example of sequential constraints, Liu et al. used a loss

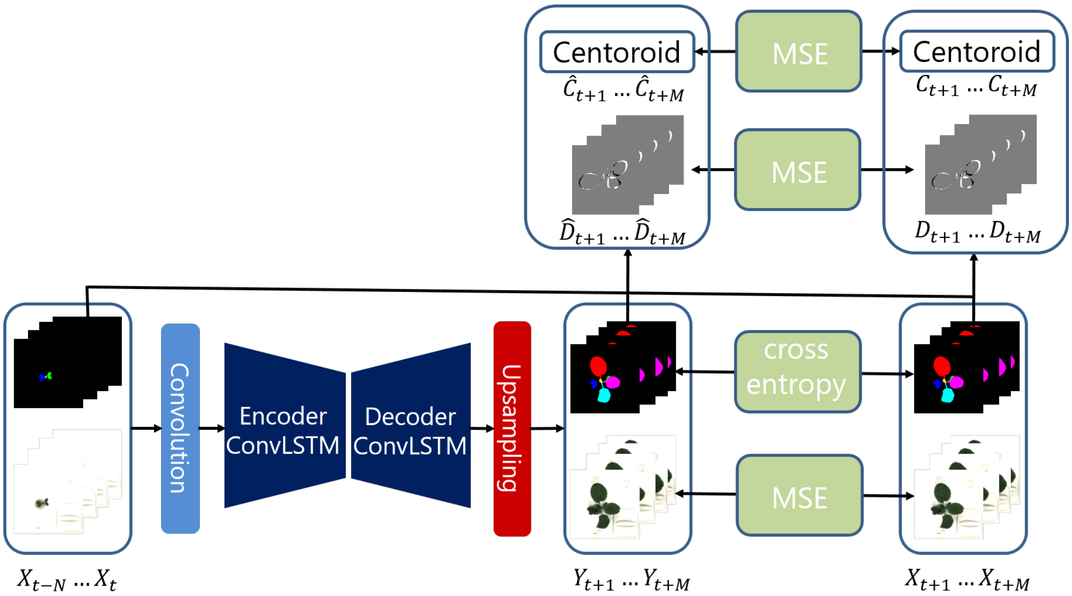


Figure 1: Overview of our proposed network plant growth prediction. Inputs and outputs are a sequence of RGB and labeled leaf images. Difference images and centroids for the training are obtained from labeled leaf images. Predictions are optimized by MSE and multi class cross entropy.

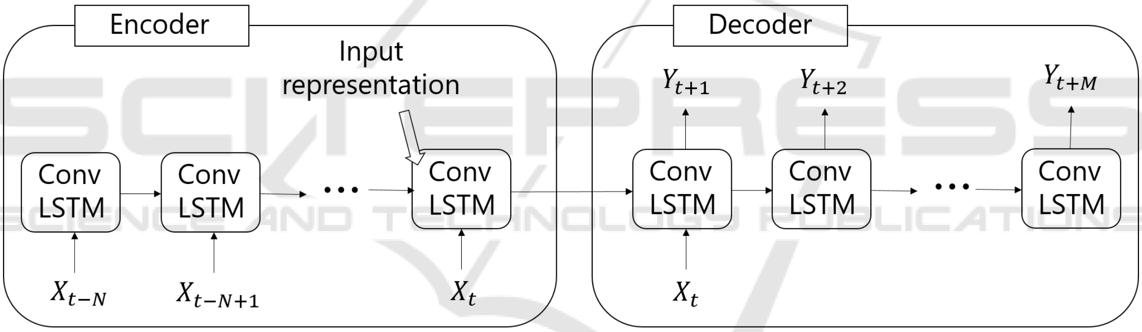


Figure 2: Detail of encoder-decoder ConvLSTM structure. The Encoder constructs the representation of inputs over time-steps and the Decoder predicts time-series future images by the representation.

function of optical flow as a constraint on motion (Liu et al., 2018). However, as mentioned above, optical flow is not available in plant growth data due to large leaf movement. To add constraint of growth expressly, we propose a difference loss that optimizes the difference between Y_t and Y_{t-1}. From the experiments, we found that this loss had a role to optimize expansion of leaves. The difference image of the t-th frame prediction $\hat{D}_t$ and corresponding ground truth D_t are defined as follows.

$$D_t = X_t^{label} - X_{t-1}^{label} \quad (3)$$

$$\hat{D}_t = Y_t^{label} - Y_{t-1}^{label} \quad (4)$$

If there is no previous prediction Y_{t-1}^{label}, X_{t-1}^{label} is used instead of the prediction. The difference loss $\mathcal{L}_{diff}$ is defined as follows.

$$\mathcal{L}_{diff}(X_t, Y_t) = MSE(D_t, \hat{D}_t) \quad (5)$$

3.4 Centroid Loss

In addition to the difference loss as a constraint of leaf appearance, we propose to use the centroid loss. This is designed as a constraint of leaf traits, which is specific for plant images. Whereas the difference loss has a role to optimize expansion of leaves, the centroid loss is has a role to optimize the movement of leaves. The centroid loss is defined by the centroid of leaves C_t and predicted $\hat{C}_t$ as follows.

$$\mathcal{L}_{cr}(X_t, Y_t) = MSE(C_t, \hat{C}_t) \quad (6)$$

We compute the centroid of leaves using image moments of labeled leaf images.

3.5 All Objective Loss

We concatenate all above described loss functions as a total objective loss. The objective loss is defined as follows.

$$\begin{aligned}\mathcal{L}(X_t, Y_t) = \lambda_{rgb}\mathcal{L}_{rgb}(X_t, Y_t) + \lambda_{label}\mathcal{L}_{label}(X_t, Y_t) \\ +\lambda_{diff}\mathcal{L}_{diff}(X_t, Y_t) + \lambda_{cr}\mathcal{L}_{cr}(X_t, Y_t)\end{aligned} \quad (7)$$

We set weights of each loss $\lambda_{rgb}, \lambda_{label}, \lambda_{diff}, \lambda_{cr}$ as 1.0, 1.0, 2.0, 0.5 respectively.

4 EXPERIMENTS

As described in the previous section, we proposed two additional loss functions to put constraints on the change between frames in the plant growth prediction task. In this section, we evaluate the effectiveness of our proposed loss functions for the plant growth prediction by using KOMATSUNA dataset (Uchiyama et al., 2017). As a quantitative evaluation score, we use the weighted coverage score (Hoiem et al., 2011; Silberman et al., 2014) between a predicted plant image result and its ground truth. The detail of the weighted coverage score is described in Section 4.3. Also, we visualized prediction of grown plant appearances as a qualitative evaluation.

4.1 Dataset

We used the KOMASTUNA dataset for evaluating our proposed network because leaf labeling over sequence was carefully performed. This dataset contains 5 plants (Komatsuna) sequential data. Each plant data consists of 60 frames captured every 4 hours from 3 viewpoints. In the experiments, we used labeled leaf images in addition to RGB images because we assume that leaf segmentation is done beforehand (Ren and Zemel, 2017; Long et al., 2015; Sakurai et al., 2018a; Li et al., 2016) and its result is used for the plant growth prediction.

The same label is assigned to each leaf through both all the frames and all the viewpoint. This label corresponds to the order of new leaves. In the dataset, the max leaf label is 8.

For the experiments, we split the data into a training dataset and a test one. We used four plant data for the training and the rest (one plant) for the testing. We used 128 × 128 image resolution, and augmented the data with 90, 180 and 270 degrees rotations.

4.2 Training Conditions

4.2.1 Overall Settings

The training dataset has 60 frames. In the experiments, we split 8 frames from the training dataset as randomly-selected inputs in each iteration for training the network. This means that it was set that $N = 7$ and $M = 8$ in Figure 1. The time of capturing 8 frames corresponds to 32 hours. However, we did not use last 8 frames as inputs for the training. If last 8 frames are contained inputs, ground truth corresponding the prediction does not exist. In total, we had 45 sequence of inputs in each dataset. 45 is calculated by excluding last 8 frames and first 7 frames for convenience of indices of frames.

In the training process, we applied dropout (Srivastava et al., 2014; Gal and Ghahramani, 2016) to the ConvLSTM. The dropout rate was set to 0.5. We used ReLU (Nair and Hinton, 2010) as an activation function in the prediction network excluding the ConvLSTM. The activation function of the ConvLSTM was hyperbolic tangent (tanh). We used Adam (Kingma and Ba, 2014) based optimization with the learning rate $\alpha = 0.0001, \beta1 = 0.5$, and $\beta2 = 0.99$. This training optimization was iterated 100000 times by using random 8 batches in each iteration.

4.2.2 Network Details

In this section, we explain the detail of the parameters in the network in Table 1. Conv2D means convolutional layers with 2D filters. Stack means the concatenation of each image along the sequential axis, and unstack means splitting along the sequential axis.

4.3 Weighted Coverage Score

As described before, we used labeled leaf images for the evaluation. Thus, the quantitative evaluation was performed for the predictions of labeled leaf images. To evaluate predictions, we employ the weighted coverage score as an evaluation metric. The weighted coverage score is computed from the overlap rate between predictions and ground truth, and the rate is weighted by the area (the number of pixels) of ground truth. The weighted coverage score (WCS) is defined as follows.

$$WCS(X,Y) = \frac{1}{\sum_i |X_i|} \sum_i |X_i| Overlap(X_t^{label}, Y_t^{label}) \quad (8)$$

$|X_i|$ denotes the number of pixels of ground truth and $Overlap(\cdot)$ means intersection over union (IoU) of between inputs.

Table 1: Parameters of prediction network. When layer is convolutional layer, shape shows filters size. When layer is images, shape shows images size. Branches are shown right side of the table.

layer	shape	layer	shape
Input (label)	128×128×9	Input (RGB)	128×128×3
Conv2D	3×3×32	Conv2D	3×3×32
Conv2D	4×4×32 2strides	Conv2D	4×4×32 2strides
Conv2D	3×3×64	Conv2D	3×3×64
Conv2D	4×4×64 2strides	Conv2D	4×4×64 2strides
Concat RGB with label	32×32×128		
Stack	8×32×32×128		
ConvLSTM(Enc)	3×3×128		
ConvLSTM(Dec)	3×3×128		
Unstack	32×32×128		
Conv2D	3×3×64		
Conv2D	4×4×64		
Upsampling	64×64×64		
Conv2D	3×3×32		
Conv2D	4×4×32		
Upsampling	128×128×32		
Conv2D	3×3×32	Conv2D	3×3×32
Conv2D	3×3×9	Conv2D	3×3×3
Softmax		tanh	
Output(label)	128×128×9	Output(RGB)	128×128×3

4.4 Results

To evaluate the effectiveness of our proposed difference loss and centroid loss for the plant growth prediction, we compared results of the experiment with or without each loss function. We had four conditions.

1. No Additional : without both difference loss and centroid loss
2. Difference : with difference loss and without centroid loss
3. Centroid : without difference loss and with centroid loss
4. Difference+Centroid : with both difference loss and centroid loss

4.4.1 Quantitative Results

We compared the weighted coverage scores of all the results of each condition described in Table 2. The leaf labels in KOMATSUNA dataset included 1 to 8 excluding background. However, the number of training data including 8 labeled leaves is few and the score is 0 in all conditions. Thus we excluded the score of leaf8 from this table and the following results.

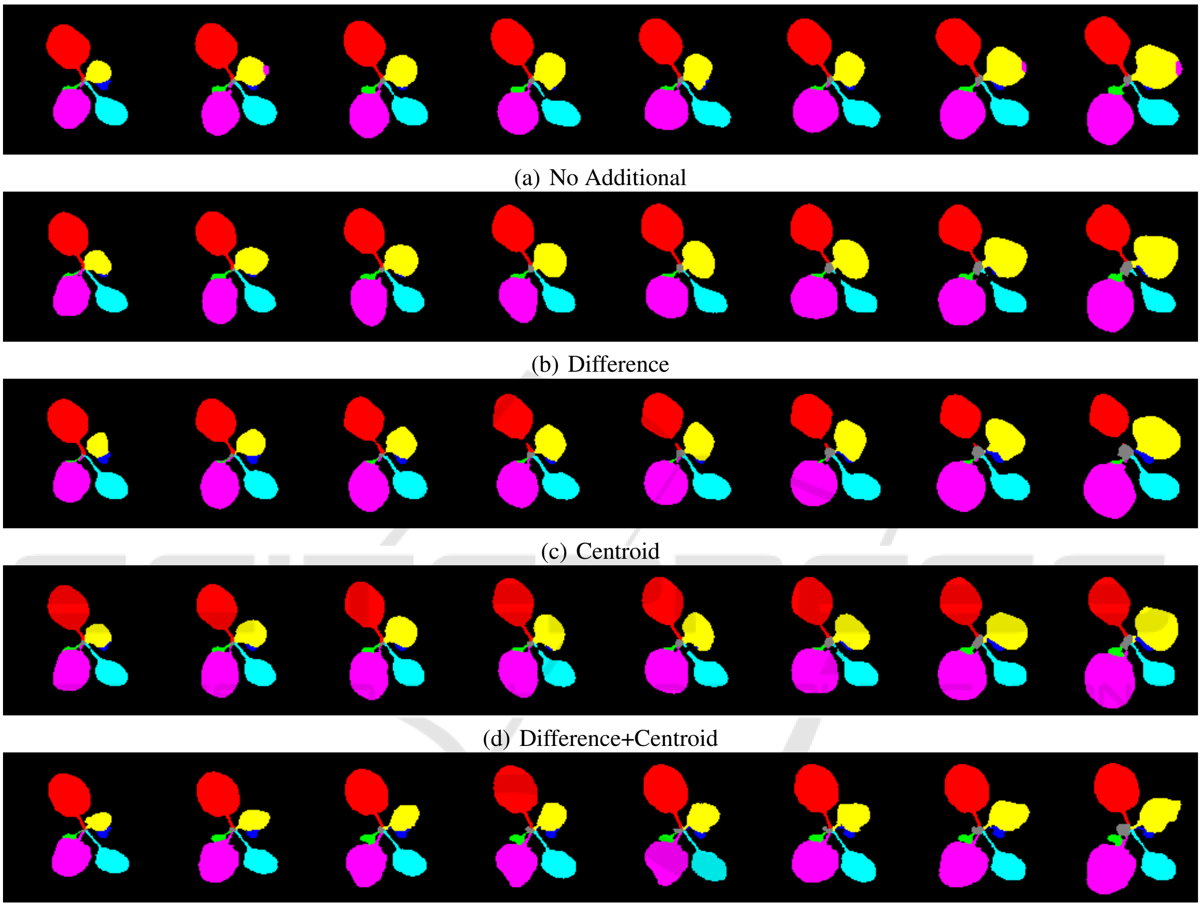
Table 2 shows the coverage score with the additional loss function tended to decrease for leaves of the earlier stage but increase for leaves of the later stage. In the earlier stage, leaves were small and shew little change between frames. On the other hand, in the later stage, leaves were large and shew great change between frames. In terms of the purpose to optimize the change for the plant growth prediction, the effectiveness of proposed loss functions were shown. Although We could see that difference loss was more effective than centroid loss, uniting difference loss with centroid loss showed the best score in many leaves containing mean coverage score.

4.4.2 Qualitative Results

Figure 3 shows the prediction of labeled leaf images Y^{label} in each condition and its ground truth X^{label}. Figure 4 shows the rgb images Y^{rgb} and X^{rgb}. We can see the prediction error in yellow leaves in results of No Additional. Such type of error disappeared by adding proposed loss. This shows proposed loss improved the visual quality of prediction. Indeed, results employed both loss function were more consistent than any other results. In prediction of RGB images, shape of leaves roughly same with corresponding labeled leaf images but all results are blurred. To improve the sharpness of rgb images, other strategy was required.

Table 2: Weighted coverage score of experimental result on KOMATSUNA dataset. Rows show score of each leaf prediction and mean over leaves. Colomns show condition of loss function.

	leaf1	leaf2	leaf3	leaf4	leaf5	leaf6	leaf7	mCov
No Additional Loss	72.5	**68.7**	67.8	72.6	66.6	61.1	32.4	67.2
Difference	72.1	67.1	67.2	73.4	**69.2**	60.7	**37.5**	67.9
Centroid	71.8	65.2	**68.5**	72.7	67.7	59.5	34.8	67.1
Difference+Centroid	**72.8**	64.7	68.2	**74.3**	69.0	**62.2**	37.2	**68.1**



(a) No Additional

(b) Difference

(c) Centroid

(d) Difference+Centroid

(e) Ground Truth

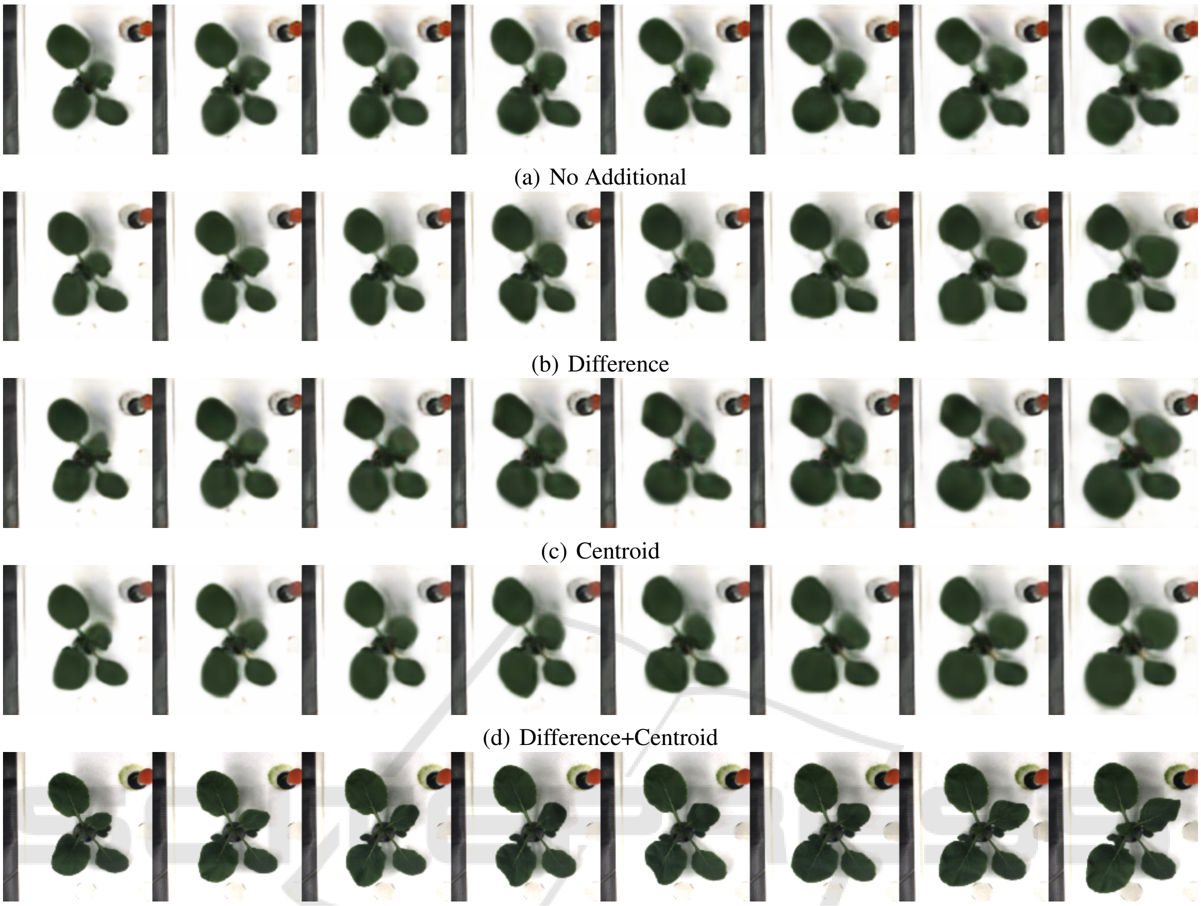
Figure 3: Predicted results of label images. Leftmost is the first frame and rightmost is the last frame. (a)-(d) show result of prediction and (e) shows its ground truth.

5 CONCLUSION

In this paper, we proposed the plant growth prediction network inspired by the future frame prediction and loss functions to optimize change of leaves between frames. We compared several conditions with/without proposed loss functions and evaluated results by the weighted coverage score of each leaf as the quantitative evaluation and the predicted appearance as the qualitative evaluation. Uniting both difference loss and centroid loss showed higher performance than the condition with no additional loss and gave the effectiveness to constrain the change of leaves.

ACKNOWLEDGEMENT

A part of this research was funded by JSPS KAKENHI grant number JP17H01768 and JP18H04117.



(a) No Additional

(b) Difference

(c) Centroid

(d) Difference+Centroid

(e) Ground Truth

Figure 4: Leftmost is the first frame and rightmost is the last frame. (a)-(d) show result of prediction and (e) shows its ground truth.

REFERENCES

Barbagallo, R. P., Oxborough, K., Pallett, K. E., and Baker, N. R. (2003). Rapid, noninvasive screening for perturbations of metabolism and plant growth using chlorophyll fluorescence imaging. *Plant Physiology*, 132(2):485–493.

Gal, Y. and Ghahramani, Z. (2016). A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*, pages 1019–1027.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8):1735–1780.

Hoiem, D., Efros, A. A., and Hebert, M. (2011). Recovering occlusion boundaries from an image. *International Journal of Computer Vision*, 91(3):328–346.

Kingma, D. P. and Ba, J. L. (2014). Adam: Amethod for stochastic optimization. In *Proc. 3rd Int. Conf. Learn. Representations*.

Ledig, C., Theis, L., Huszar, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4681–4690.

Li, K., Hariharan, B., and Malik, J. (2016). Iterative instance segmentation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Li, L., Zhang, Q., and Huang, D. (2014). A review of imaging techniques for plant phenotyping. *Sensors*, 14(11):20078–20111.

Liu, W., Luo, W., Lian, D., and Gao, S. (2018). Future frame prediction for anomaly detection a new baseline. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Long, J., Shelhamer, E., and Darrell, T. (2015). Fully convolutional networks for semantic segmentation. In

Proceedings of the IEEE conference on computer vision and pattern recognition, pages 3431–3440.

Luc, P., Couprie, C., Chintala, S., and Verbeek, J. (2016). Semantic segmentation using adversarial networks. In *NIPS Workshop on Adversarial Training*.

Mathieu, M., Couprie, C., and LeCun, Y. (2016). Deep multi-scale video prediction beyond mean square error. *ICLR*.

Moriyuki, S. and Fukuda, H. (2016). High-throughput growth prediction for lactuca sativa l. seedlings using chlorophyll fluorescence in a plant factory with artificial lighting. *Frontiers in plant science*, 7:394.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814.

Ranzato, M., Szlam, A., Bruna, J., Mathieu, M., Collobert, R., and Chopra, S. (2014). Video (language) modeling: a baseline for generative models of natural videos. *CoRR*, abs/1412.6604.

Ren, M. and Zemel, R. S. (2017). End-to-end instance segmentation with recurrent attention. In *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA*, pages 21–26.

Sakurai, S., Uchiyama, H., Shimada, A., Arita, D., and ichiro Taniguchi, R. (2018a). Two-step transfer learning for semantic plant segmentation. In *Proceedings of the 7th International Conference on Pattern Recognition Applications and Methods - Volume 1: ICPRAM,*, pages 332–339. INSTICC, SciTePress.

Sakurai, S., Uchiyama, H., Shimada, A., Arita, D., and Taniguchi, R. (2018b). Two-step transfer learning for semantic plant segmentation. In *7th International Conference on Pattern Recognition Applications and Methods (ICPRAM2018)*.

Silberman, N., Sontag, D., and Fergus, R. (2014). Instance segmentation of indoor scenes using a coverage loss. In *European Conference on Computer Vision*, pages 616–631. Springer.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Srivastava, N., Mansimov, E., and Salakhudinov, R. (2015). Unsupervised learning of video representations using lstms. In *International conference on machine learning*, pages 843–852.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

Ubbens, J. R. and Stavness, I. (2017). Deep plant phenomics: a deep learning platform for complex plant phenotyping tasks. *Frontiers in plant science*, 8:1190.

Uchiyama, H., Sakurai, S., Mishima, M., Arita, D., Okayasu, T., Shimada, A., and Taniguchi, R.-i. (2017). An easy-to-setup 3d phenotyping platform for komatsuna dataset. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*.

Villegas, R., Yang, J., Hong, S., Lin, X., and Lee, H. (2017). Decomposing motion and content for natural video sequence prediction. *ICLR*.

Walter, A., Liebisch, F., and Hund, A. (2015). Plant phenotyping: from bean weighing to image analysis. *Plant methods*, 11(1):14.

Xingjian, S., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c. (2015). Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in neural information processing systems*, pages 802–810.

Zaidi, M., Murase, H., and Honami, N. (1999). Neural network model for the evaluation of lettuce plant growth. *Journal of agricultural engineering research*, 74(3):237–242.