



Information Acquisition Method of Tomato Plug Seedlings Based on Cycle-Consistent Adversarial Network



Yong Zhang^{*ID}

School of Information and Electronic Engineering, Shandong Technology and Business University, 264005
Yantai, China

* Correspondence: Yong Zhang (zzxd@sdtbu.edu.cn)

Received: 01-16-2023

Revised: 02-18-2023

Accepted: 03-12-2023

Citation: Y. Zhang, "Information acquisition method of tomato plug seedlings based on cycle-consistent adversarial network," *Acadlore Trans. Mach. Learn.*, vol. 2, no. 1, pp. 46-54, 2023.
<https://doi.org/10.56578/ataiml020105>.



© 2023 by the authors. Licensee Acadlore Publishing Services Limited, Hong Kong. This article can be downloaded for free, and reused and quoted with a citation of the original published version, under the CC BY 4.0 license.

Abstract: In order to solve the interference caused by the overlapping and extrusion of adjacent plug seedlings, accurately obtain the information of tomato plug seedlings, and improve the transplanting effect of automatic tomato transplanters, this study proposes a seedling information acquisition method based on Cycle-Consistent Adversarial Network (CycleGAN). CycleGAN is a generative unsupervised deep learning method, which can realize the free conversion of the source-domain plug seedling image and the target-domain plug label image. It collects more than 500 images of tomato plug seedlings in different growth stages as a collection image set; follows certain principles to label the plug seedling images to obtain a label image set, and uses two image sets to train the CycleGAN network model. Finally, the trained model is used to process the images of tomato plug seedlings to obtain their label images. According to the labeling principle, the correct rate of model recognition is between 91% and 97%. The recognition results show that the CycleGAN model can recognize and judge whether the seedlings affected by the adjacent seedling holes are suitable for transplanting, so the application of this method can greatly improve the intelligence level of the automatic tomato transplanters.

Keywords: Deep learning; Cycle-Consistent Adversarial Networks; Tomato transplanter; Plug seedling recognition; Label image

1. Introduction

China is one of the largest tomato producers in the world. Plug seedling transplanting is the main planting method of tomato production. With the increase of planting amount and the rise of labor costs, mechanical transplanting is being used more and more, but at present most of the transplanters used are semi-automatic transplanters for manual feeding of seedlings, which cannot fundamentally solve such problems as high labor intensity, low transplanting efficiency, and low transplanting accuracy. The automatic tomato transplanter automatically grabs the seedlings from the plug tray, and then plants them in the field. With the advantages of high transplanting efficiency and low labor intensity, it has become the research focus of relevant institutions. However, because of influencing factors such as seed germination rate and seedling raising environment, there will be phenomena such as lack of seedlings and weak seedlings in the plug tray. If automatic transplanters cannot identify the seedling holes that are not suitable for transplanting, and still transplant the seedling from the seedling hole regularly, the transplanting effect of automatic transplanters will be greatly affected, resulting in an increase in the rate of missed planting. Therefore, identifying and obtaining the tomato seedling information in each seedling hole of the plug tray is a key technology to improve the transplanting effect of automatic tomato transplanters.

Scholars and related institutions at home and abroad have conducted research on how to obtain information on plug seedlings. Tai et al. [1] developed an automatic transplanter assisted by machine vision. Based on the gray information of the image, the segmentation threshold was obtained through sampling to determine whether the seedling hole is empty. Ryu et al. [2] developed a seedbed transplanting robot with a vision system, which uses a pre-defined value for image segmentation to determine empty seedling holes so as to reduce transplanting time. The Futura high-end automatic transplanter from Ferrari, Italy, uses photoelectric technology to scan the plug

seedlings to determine whether there is a lack of seedlings. Prasanna Kumar and Raheman [3] designed a vegetable transplanter based on a walking tractor, and Li et al. [4] and Yang et al. [5]. and Rahul [6] carried out kinematic analysis and research on the pot seedling picking mechanism and control system. The above research is involved with two types of detection methods. One is photoelectric sensor detection, which can only detect the lack of seedling information, with single function; and it is easy to be interfered by the size of the seedlings, the growth angle, and the seedlings in adjacent seedling holes, with a low level of intelligence. The other is the image processing method. The above image processing methods are all based on image segmentation. After the image segmentation, the continuity information of some stems and leaves of the seedlings will be lost. When the stems and leaves of adjacent seedling holes overlap and squeeze each other, there will be misjudgment phenomenon. Tomato seedling plants are relatively large, and the stems and leaves of seedlings in adjacent seedling holes have begun to overlap at about 20 days during the seedling cultivation period. In order to obtain the information of tomato plug seedlings correctly, it's necessary to adopt more effective methods.

In recent years, Generative Adversarial Network (GAN) [7-9] has been widely used in image generation, image style transfer, image repair and other fields. It is a generative unsupervised deep learning model that can use a small amount of training data to generate target data sets, making up for the problem of insufficient training data [10]. One of the main disadvantages of GAN is that the training process is unstable. For this reason, various improved models have been proposed, such as Conditional Generative Adversarial Nets (CGAN) [11] and Deep Convolutional Generative Adversarial Networks (DCGAN) [12]. CycleGAN [13], proposed as a solution to image-to-image conversion, can generate images in both directions and realize the mutual conversion of images in any two domains. In this paper, CycleGAN is used to train the images of plug seedlings and label images, so as to realize the conversion from images of plug seedlings to label images. Through the labeling principle, it is possible to directly judge whether each seedling hole is suitable for transplanting. The ring network structure of CycleGAN makes it possible for the generator and the discriminator to have a stronger learning and discrimination ability, thus extracting more image characteristic information of plug seedlings and solving the mutual interference of seedlings in adjacent seedling holes.

2. Structure and Working Principle of Intelligent Automatic Tomato Transplanters

Intelligent automatic tomato transplanters can transmit tomato plug seedlings, pick seedlings by the mechanical picking arm, feed seedlings, plant seedlings and complete other actions to realize fully automatic tomato transplanting. Its machine vision system can judge whether the plug seedlings are suitable for transplanting, thus improving the transplanting effect.

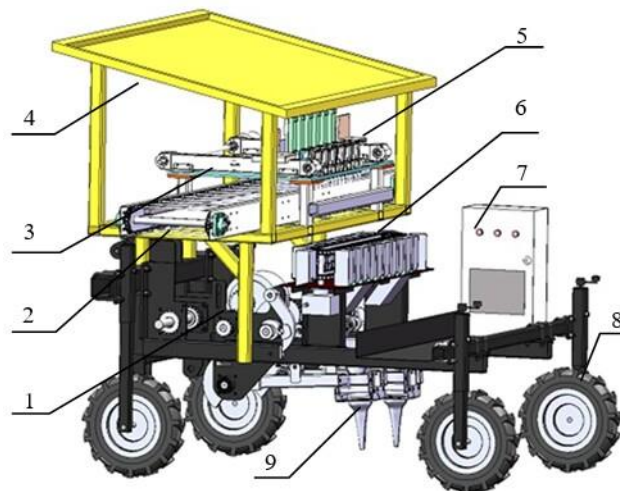


Figure 1. Structure diagram of automatic seedling transplanter: 1. Transmission mechanism; 2. Seedling feeding mechanism; 3. Industrial camera; 4. Seedling tray frame; 5. Seedling picking mechanism; 6. Transverse seedling throwing mechanism; 7. Electrical control box; 8. Limit wheel; 9. Five-pole planting mechanism

The automatic tomato transplanter is shown in Figure 1. It mainly includes a vision system, a seedling feeding mechanism, a seedling picking mechanism, a transverse seedling throwing mechanism, a five-pole planting mechanism, a transmission mechanism, and a control box. The vision system consists of an industrial camera, an industrial control computer and a controller. The industrial camera is installed directly above the tray, and the industrial control computer and the controller are placed in the control box. During the transplanting operation, the operator removes the tray from the tray frame on the console and places it on the seedling feeding mechanism,

which is driven by a stepping motor. The precise positioning of the tray is realized under the control of the controller. Three trays can be placed on the mechanism at the same time, of which the middle one stops directly under the industrial camera and is in a waiting state, and the rightmost one stops directly under the seedling picking mechanism for transplanting. The seedling picking mechanism includes six seedling picking hands, which reciprocate under the drive of the servo motor to take out seedlings from the tray and put them into the transverse seedling throwing mechanism. The transverse seedling throwing mechanism moves to the seedling throwing position and puts seedlings into the five-pole planting mechanism, which plants the seedlings into the field. When the tray in the middle of the seedling feeding mechanism is in the waiting state, pictures, taken by the camera triggered by a command issued by the controller, are transmitted to the industrial control computer for processing to obtain such information as lack of seedlings and weak seedlings in the tray, which is transmitted to the controller. Therefore, during transplantation of seedlings in the tray, the movement of seedling picking hands in the seedling picking mechanism is controlled to give up grabbing seedlings unsuitable for transplanting, and an audible and visual alarm message is sent to remind the operator of replenishing seedlings in time [14].

3. Information Acquisition Method of Plug Seedlings Based on CycleGAN

In this paper, CycleGAN is used to obtain the information of plug seedlings. The processing flow includes two stages: the first stage is to use different colors to mark the seedling status of each hole in the image of the plug seedling. Green means that the seedlings in the hole are suitable for transplanting, while yellow indicates that the seedling hole is short of seedlings or the seedlings are weak and not suitable for transplanting, and the edge of the seedling hole is indicated by red. So the marked picture is the label image. In the second stage, the CycleGAN model is trained with images of plug seedlings and label images, and finally it is possible to realize the mutual conversion between the collected images of the plug seedlings and the label images. Based on label images and the labeling principle, it is easy to judge whether the plug seedlings are suitable for transplanting.

3.1 Network Model for Generating Label Images of Plug Seedlings

CycleGAN contains two generators and two discriminators with parameters shared. The ring structure realizes the function of cycle consistency. It performs a two-step transformation on source domain images: first, it tries to map source domain images to the target domain through the first generator, and then return images to the source domain through the second generator to obtain secondary generated images. The generator network is not only responsible for generating the target image, but also constantly adjusts itself to make the generated image more like the real data set; while the discriminant model is responsible for identifying the authenticity of generated images or real images. If the image is false, the output label is 0, otherwise it is recorded as 1. According to the label, it is judged whether the image comes from the real sample or the generative model, so as to guide the generation process of the generative model. By matching the generator and the discriminator, the quality of the generated image can be greatly improved.

The collected image domain of plug seedlings is set as X , and the label image domain as Y . The essence of using the CycleGAN network model to obtain label images in this paper is to allow the network to realize the two mappings of $X \rightarrow Y'$ and $Y \rightarrow X'$ with the ring structure. Figure 2 shows the principle of using CycleGAN to obtain label images. It can be seen from Figure 2 that the CycleGAN network model includes two generative models of G_X and G_Y and two discriminant models of D_X and D_Y . The generator G_X is defined to learn the mapping from $X \rightarrow Y'$ and the generator G_Y to learn the mapping from $Y \rightarrow X'$. The discriminator D_X is not only used to judge whether the image is from the real label image X or from the generated $G_Y(Y)$, but also to judge whether the image is the image $G_Y(G_X(X))$ generated by the generator G_Y or the real label image; the discriminator D_Y is used to judge whether the data comes from the real collected image Y or the generated $G_X(X)$, and also to judge whether the image is the image $G_X(G_Y(Y'))$ generated by G_X or the real collected image.

The specific processing progress of network training can be described as follows: After images in the label image set X are input into the first generative model G_X , G_X generates an image similar to Y , denoted as \tilde{Y} . Then the discriminant model D_Y classifies \tilde{Y} , judges whether \tilde{Y} comes from the real collected data set or the image generated by the generative model, and feedbacks the result to G_X so as to strengthen G_X and make G_X generate more realistic images. Therefore, under the supervision of D_Y , the gap between \tilde{Y} and the image in the data set Y can be continuously narrowed, that is, the generated color image should be as similar as possible to the real collected data set Y . After \tilde{Y} is input into the second generative model G_Y again, another new label image X' is generated, which should be as similar as possible to the original label image data set X . Conversely, after the image in Y is input into G_Y , the image \tilde{X} generated should be as similar as possible to the image in X under the supervision of the discriminant model D_X , and the new collected image Y' generated by G_X from \tilde{X} should be as similar as possible to the image in Y . That is to say, the generative model should not only transform X into Y , but also have the ability to restore from Y to X , and vice versa. This is the special ring structure of this model.

The fundamental principle of GAN is the mutual game between the generative model and the discriminant

model. In the model of this paper, D_Y trains G_X , and D_X trains G_Y , so that the two generative models can generate more and more realistic images. And the generative models respectively train the discriminant model in return to continuously enhance the resolution ability. The generation process of the new label image X' and the new collected image Y' , taken as a verification process, is to verify whether the generative model is well trained to generate enough just-as-good fake images. After the model training is completed, and the image of the plug seedlings in the collected data set Y is input, the generative model G_Y can generate the corresponding label image, that is, transform Y into X' , and vice versa, so that it is possible to realize the transformation from the collected data set to the label data set. Although the transformation process from X to Y must exist for training the model, for practical needs, the output result only retains the transformation result from Y to X , that is \tilde{X} , to realize the function of obtaining the label image.

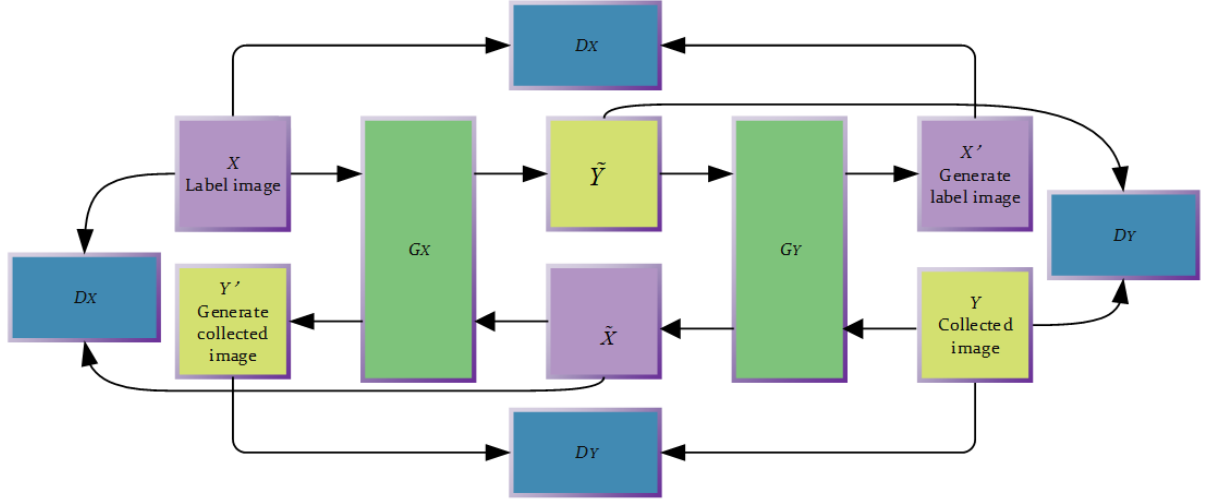


Figure 2. Principle of capturing label images by CycleGAN

3.1.1 Generator network

The generator network is to make the output as close as possible to the real samples. In this paper, two generator networks G_X and G_Y are used, and the network structures of G_X and G_Y are consistent. Figure 3 shows the structure of the generator network, which has 17 layers, including: 1 input layer, 3 convolution layers, 9 residual blocks, 2 deconvolution layers, 1 1×1 convolution layer and 1 output layer. The purpose of the convolution layer is to extract different features of the input image, and the output features of the convolution layer can be obtained by formula (1):

$$F = \text{ReLU}(X \times W + b) \quad (1)$$

where, X represents the image matrix, W represents the convolution kernel, \times represents the convolution computation, and b represents the bias value.

The activation function used here is a rectified linear unit (ReLU) [15]. It is faster and more accurate to train a convolutional neural network with the ReLU function. An image with a size of 111×76 is input into the generative model. The image features are extracted through three convolution layers, and then input into the residual block. The structure of the residual block is shown in Figure 4. Each residual block layer contains 2 convolution layers, 1 Instance Normalization (IN) layer [16] and an activation function ReLU layer. The residual block directly adds part of the input to the output at the end, which can realize image conversion while protecting the integrity of the original image information. The residual block used here can not only solve the degradation problem caused by network deepening [17], improving the recognition accuracy of the model, but also enhance the translation ability of the network. Similar to convolution, deconvolution [18] uses the filter transposed in the convolution process. The feature vector of the image is restored to low-level features by the deconvolution layer. Compared to convolution, deconvolution processes the opposite operation in the direction and back propagation of the neural network architecture. The function of deconvolution is to achieve the effect of reconstructing the input from the output through reverse training. In this paper, two layers of deconvolution are used. The final convolution layer is used to convert the input data into a feature map with a dimension of 1×1 , and finally the ReLU activation layer is used to process non-linear change to the feature map with a dimension of 1×1 , thus generating a new image.

In order to achieve the normalization operation of a single image, IN is used for normalization after each convolution layer. Unlike BN (Batch Normalization) [19], IN is to obtain the mean and the standard deviation of all pixels of a single image, and the output calculation statistics are only related to the current picture itself. While

BN calculates the mean and the standard deviation of all pixels in all pictures in the entire batch, and the output calculation statistics will be affected by other samples in the batch. Therefore, use of IN can improve the quality of generated images on the basis of accelerating model convergence and preventing gradient explosion.

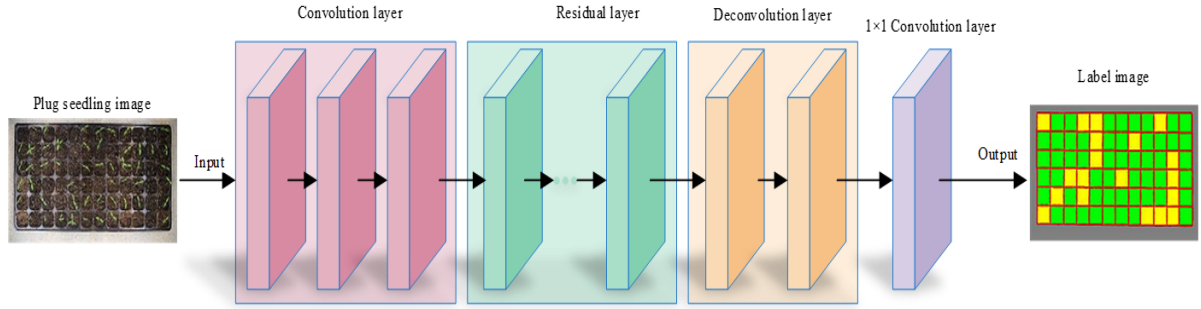


Figure 3. Structure of the generator network

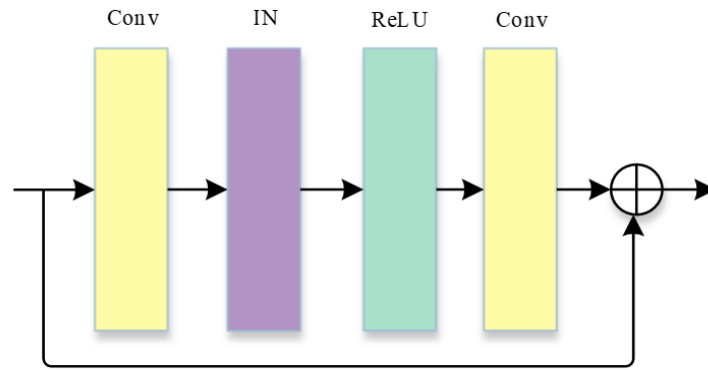


Figure 4. Residual block structure

3.1.2 Discriminator network

The discriminator network is to supervise the generator and judge whether the pictures are generated by the generator or real pictures. Figure 5 shows the network structure of the discriminant model, which is a fully convolutional network composed of 7 convolution layers. The image features are extracted through 6 convolution layers, and then the judgment result is output through the last convolution layer so as to realize the judgment function. Except for the first and the last layers, other convolution layers all adopt LReLU for normalization.

The output result of the discriminant model is a one-dimensional vector. For the input label image, if the discriminator judges that the picture is an image generated by the generative model, it outputs 0; if the discriminator judges that the picture is a real image from the sample set, it outputs 1.

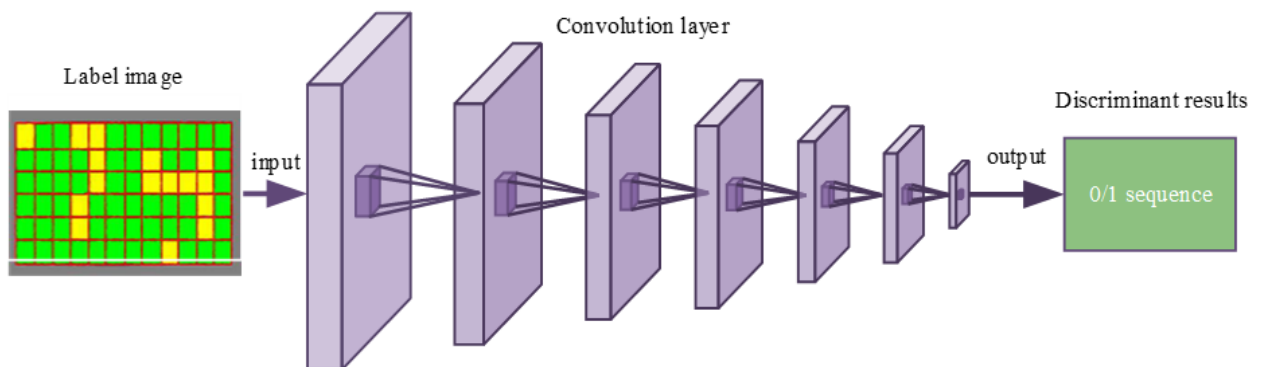


Figure 5. Discriminant model network structure

3.2 Loss Function

In order to better train the neural network, and to convert the samples in the X space to the samples in the Y space with CycleGAN, according to the generator G_X and the discriminator D_Y in this paper, a generative discriminator loss function is constructed. When the back error propagation of the real samples and the generated samples are used to guide the learning of the generator G_X and D_Y network parameters, where the samples $x \in X$ and $y \in Y$, the function is expressed as:

$$L_{G_X}(G_X, D_Y, X, Y) = E_{y \sim P_{data}(y)}[\log D_Y(y)] + E_{x \sim P_{data}(x)}[\log(1 - D_Y(G_X(x)))] \quad (2)$$

Conversely, when the CycleGAN training process needs to convert the samples in the Y space to the samples in the X space, and construct a generative discriminator loss function to guide the generator G_Y and D_X network parameter learning, the expression is:

$$L_{G_Y}(G_Y, D_X, Y, X) = E_{x \sim P_{data}(x)}[\log D_X(x)] + E_{y \sim P_{data}(y)}[\log(1 - D_X(G_Y(y)))] \quad (3)$$

According to CycleGAN's description of the cycle consistency loss, that is, the image X is mapped to Y , and Y should be mapped back to X at the same time, the cycle consistency loss between the original image X and the mapped image is calculated, and the loss function is defined as:

$$L_c(G_X, G_Y, X, Y) = E_{x \sim P_{data}(x)}[\|G_Y(G_X(x)) - x\|_1] + E_{y \sim P_{data}(y)}[\|G_X(G_Y(y)) - y\|_1] \quad (4)$$

The total loss function consists of 3 parts: the generative discriminator loss function from X to Y , the generative discriminator loss function from Y to X , and the cycle consistency loss function. The total loss expression is:

$$L(G_X, G_Y, D_X, D_Y) = L_{G_X}(G_X, D_Y, X, Y) + L_{G_Y}(G_Y, D_X, Y, X) + \lambda L_c(G_X, G_Y, X, Y) \quad (5)$$

where, λ is a parameter that controls the weights for generative discriminator loss and cycle consistency loss objects.

3.3 Network Training

The network is trained through the loss function. In the training phase, the Adam optimizer is used to run 60 epochs, the learning rate of the first 30 is 1e-4, and the learning rate update strategy of the last 30 is:

$$\alpha_{epoch} = \alpha \times \frac{60 - epoch}{30} \quad (6)$$

where, α_{epoch} represents the learning rate of the current epoch, and α represents the initial learning rate.

The whole training process can be described as: first, the label image is input into G_X and processed by the generator network to generate \tilde{Y} , then the discriminant model D_Y is used to distinguish \tilde{Y} and the collected seedling tray image Y . Next, \tilde{Y} is input into the second generative model G_Y to generate an image set X' that is similar to the label image set X . Conversely, the collected seedling tray image set Y is input into G_Y to output \tilde{X} , and the discriminant model D_X is used to distinguish \tilde{X} and label images, and then \tilde{X} is input into G_X to generate an image Y' similar to the collected seedling tray image. Only when the differences between \tilde{X} and X , \tilde{Y} and Y , X' and X , and Y' and Y reach the minimum value, that is to say, when D_X and D_Y cannot distinguish the real data set from the generated data set, the whole network training process shall finish.

4. Experiment and Result Analysis

4.1 Experimental Data

As the information acquisition method of plug seedling based on CycleGAN requires image samples for training, three trays of tomato seedlings were cultivated in an artificial greenhouse. The temperature of the greenhouse was kept at 25°C during the day and 15°C at night. The seedling substrate is composed of peat, vermiculite and nutrient soil, and the seedling tray adopts a 12×6 hole tray. The image acquisition device is shown in Figure 6, which consists of a frame with the height of 50cm and a camera of Canon 550D SLR with 18 million pixels. The first tray of tomato seedlings was sown on October 8, 2018, and the remaining two trays were sown two days apart. On

October 19, the shoots were taken at 9 angles such as directly above the seedling tray and in the middle, and under different light conditions. More than 500 images of tomato seedling trays were captured during the seedling stage of 12~30d. All images are compressed into the images with a pixel size of 256×256, which are used as a tomato seedling tray image sample set. In addition, label image sample sets are generated by labeling according to the labeling principle, which are used for the training of the CycleGAN network.

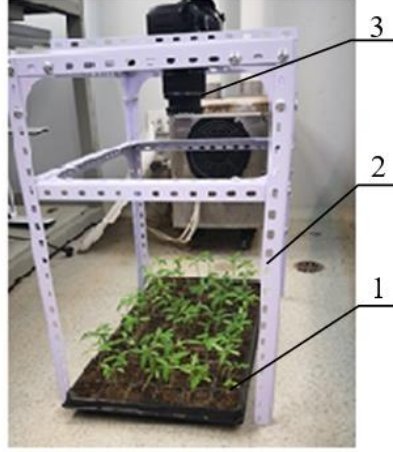


Figure 6. Image acquisition device: 1. Tomato seedlings; 2. Frame; 3. Camera

4.2 Experimental Environment

The environment used for the experimental operation is completed on a Dell Vostro 3470-R1328R desktop computer, with the configuration of 8th generation Core i7 Processor, 8G, and Windows10 operating system. The programming software used in the experiment is Anaconda, the framework is TensorFlowV1.2, and the programming language is Python3.5.

4.3 Evaluation Indicators

Recognition accuracy is an important index to evaluate the identification of plug seedlings, and it is calculated by the following formula:

$$P = \frac{A_0}{A} \times 100\% \quad (7)$$

where, P is the recognition accuracy rate, A is the total number of seedling holes on the plug tray (take 72), A_0 is the number of correctly identified hole information on the plug seedlings, and the number correctly recognized consists of two parts: for one part, the seedlings in holes in the actual image are suitable for transplanting, and the recognition results also show that they are suitable for transplanting; for the other part, seedlings in holes are lacked, lodged or weak in the real image, and the recognition results show that they are not suitable for transplanting. The higher the accuracy rate is, the stronger the recognition ability of the model, and the higher the probability that the plug seedling information is recognized correctly.

4.4 Experimental Results and Analysis

The CycleGAN network recognizes tomato seedling images after image training samples are trained. Figure 7 shows 8 representative recognition results, which are composed of actual collected images and recognition results. The principle of displaying the recognition results is consistent with the labeling principle of the label image, of which green means it is suitable for transplanting, and yellow means it is not suitable for transplanting.

Subgraphs (a) to (d) of Figure 7 represent the early stages of tomato seedling cultivation. At this time, the stems and leaves of tomato seedlings are basically in their respective seedling holes, the seedlings in adjacent seedling holes have little influence, and the recognition accuracy rate is 94%~97%. The main reason of the wrong identification of seedling holes is that the seedlings just break through the soil and are small in size, and it is relatively close in color to the surrounding seedling substrate. In subgraphs (e) to (h) of Figure 7, the leaves of the seedlings in adjacent seedling holes begin to overlap, the recognition accuracy is 91%~97%, and the misidentified

seedling holes are mainly concentrated in a row near the edge. As seedlings are artificially sown, tomato seedlings of the row are not sown in the middle of the seedling hole, but leans to the outside, so most of the stems and leaves of the seedlings grow outside the plug seedling tray. While the label image used for training is bounded by the outer edge of the plug seedling tray, resulting that the judgment information of the seedlings in this part is lost.

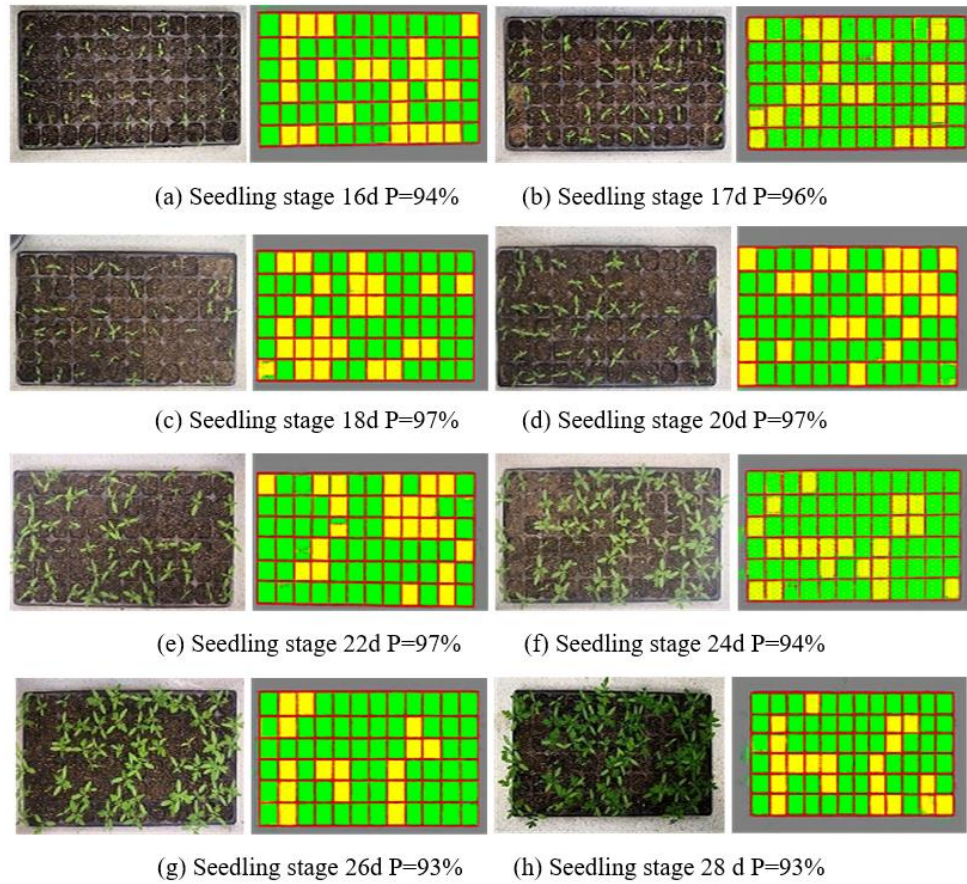


Figure 7. Recognition results

5. Conclusions

This paper designs an information acquisition method of plug seedlings based on CycleGAN. The model of this method is a ring network structure, including two generators and two discriminators with parameters shared, which is equivalent to an organic combination of two mirror-symmetric one-way GANs. This method eliminates the requirement of image matching in the target domain, which can greatly simplify the work in the training data preparation stage. The special ring structure enables the generator network to realize the bidirectional mapping between the tomato seedling image and the label image, and the discriminator network guides the generation process of the generator network. And through the matching of the generator and the discriminator, the generated image can be greatly improved. The experimental results show that, compared with the previous acquisition methods, the information acquisition method of plug seedlings based on CycleGAN can make more comprehensive use of the feature information of plug seedling images. And the generator model can obtain overlapping and extrusion features of seedling stems and leaves in adjacent holes, so as to identify and judge whether seedlings in the seedling holes affected by the surrounding seedling holes, and whether they are suitable for transplanting, thus greatly improving the intelligence level of automatic tomato transplanters.

Data Availability

The data used to support the research findings are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflict of interest.

References

- [1] Y. W. Tai, P. P. Ling, and K. C. Ting, "Machine vision assisted robotic seedling transplanting," *Trans. ASAE*, vol. 37, no. 2, pp. 661-667, 1994. <https://doi.org/10.13031/2013.28127>.
- [2] K. H. Ryu, G. Kim, and J. S. Han, "Development of a robotic transplanter for bedding plants," *J. Agric. Eng. Res.*, vol. 78, no. 2, pp. 141-146, 2001. <https://doi.org/10.1006/jaer.2000.0656>.
- [3] G. V. Prasanna Kumar and H. Raheman, "Development of a walk-behind type hand tractor powered vegetable transplanter for paper pot seedlings," *Biosyst. Eng.*, vol. 110, no. 2, pp. 189-197, 2011. <https://doi.org/10.1016/j.biosystemseng.2011.08.001>.
- [4] H. Li, W. B. Cao, S. F. Li, W. Fu, and K. Q. Liu, "Kinematic analysis and test on automatic pick-up mechanism for chili plug seedling," *Trans. Chin. Soc. Agric. Eng.*, vol. 31, no. 23, pp. 20-27, 2015.
- [5] Q. Yang, L. Xu, X. Y. Shi, A. Ibrar, H. P. Mao, J. P. Hu, and L. H. Han, "Design of seedlings separation device with reciprocating movement seedling cups and its controlling system of the full-automatic plug seedling transplanter," *Comput. Electron. Agric.*, vol. 147, pp. 131-145, 2018. <https://doi.org/10.1016/j.compag.2018.02.004>.
- [6] K. Rahul, H. Raheman, and V. Paradkar, "Design and development of a 5R 2DOF parallel robot arm for handling paper pot seedlings in a vegetable transplanter," *Comput. Electron. Agric.*, vol. 166, Article ID: 105014, 2019. <https://doi.org/10.1016/j.compag.2019.105014>.
- [7] I. J. Goodfellow, J. Pouget-Adadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative Adversarial Nets," In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, Cambridge, MA, United States, December 2014, USA: MIT Press, pp. 2672-2680.
- [8] A. Krizhevsky, I. Sutskever, and G. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84-90, 2017. <https://doi.org/10.1145/3065386>.
- [9] S. Wen, N. W. Shen, J. T. Zhang, Y. B. Lan, J. Han, X. C. Yin, Q. Y. Zhang, and Y. F. Ge, "Single-rotor UAV flow field simulation using generative adversarial networks," *Comput. Electron. Agric.*, vol. 167, Article ID: 105004, 2019. <https://doi.org/10.1016/j.compag.2019.105004>.
- [10] M. L. Zhou, Y. Y. Shan, X. L. Xue, and D. Q. Yin, "Theoretical analysis and development of a mechanism with punching device for transplanting potted vegetable seedlings," *Int J. Agric. Biol. Eng.*, vol. 13, no. 4, pp. 85-92, 2020. <https://doi.org/10.25165/j.ijabe.20201304.5404>.
- [11] M. N. Islam, M. Z. Iqbal, M. Ali, M. Chowdhury, M. S. N. Kabir, T. Park, Y. J. Kim, and S. O. Chung, "Kinematic analysis of a clamp-type picking device for an automatic pepper transplanter," *Agriculture*, vol. 10, no. 12, 2020. <https://doi.org/10.3390/agriculture10120627>.
- [12] O. J. Jorg, M. Sportelli, M. Fontanelli, C. Frascioni, M. Raffaelli, and G. Fantoni, "Design, development and testing of feeding grippers for vegetable plug transplanters," *AgriEngineering*, vol. 3, no. 3, pp. 669-680, 2021. <https://doi.org/10.3390/agriengineering3030043>.
- [13] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, Haifa Israel, June 21-24, 2010, Israel: Omnipress, pp. 807-814.
- [14] T. M. Quan, T. Nguyen-Duc, and W. K. Jeong, "Compressed sensing MRI reconstruction using a generative adversarial network with a cyclic loss," *IEEE Trans. Med. Imaging*, vol. 37, no. 6, pp. 1488-1497, 2018. <https://doi.org/10.1109/TMI.2018.2820120>.
- [15] Y. P. Hu, E. Gibson, T. Vercauteren, H. U. Ahmed, M. Emberton, C. M. Moore, J. A. Noble, and D. C. Barratt, "Intraoperative organ motion models with an ensemble of conditional generative adversarial networks," In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Quebec City, QC, Canada, September 10-14, 2017, Springer, Cham, pp. 368-376. https://doi.org/10.1007/978-3-319-66185-8_42.
- [16] M. D. Zeiler, D. Krishnan, G. W. Taylor, and R. Fergus, "Deconvolutional networks," In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Francisco, CA, USA, June 13-18, 2010, USA: IEEE, pp. 2528-2535. <https://doi.org/10.1109/CVPR.2010.5539957>.
- [17] E. A. Smirnov, D. M. Timoshenko, and S. N. Andrianov, "Comparison of regularization methods for imagenet classification with deep convolutional neural networks," *AASRI Proc.*, vol. 6, pp. 89-94, 2014. <https://doi.org/10.1016/j.aasri.2014.05.013>.
- [18] S. Kida, S. Kaji, K. Nawa, T. Imae, T. Nakamoto, S. Ozaki, T. Ohta, Y. Nozawa, and K. Nakagawa, "Visual enhancement of Cone-beam CT by use of CycleGAN," *Medical Physics*, vol. 47, no. 3, pp. 998-1010, 2019. <https://doi.org/10.1002/mp.13963>.
- [19] A. Koutsoukas, K. J. Monaghan, X. Li, and J. Huan, "Deep-learning: Investigating deep neural networks hyper-parameters and comparison of performance to shallow methods for modeling bioactivity data," *J. Cheminform.*, vol. 9, no. 1, 2017.