

Research on the Method of Rice Seed Density Detection in Rice Seedling Tray Based on MCNN

Xiangwu Deng*, Dacheng Liu, Ruhui Chen, Sheng Xie, Ziqin Hong

College of Electronic Information Engineering, Guangdong University of Petrochemical Technology, Maoming, 525000, China

*Corresponding author: Xiangwu Deng

Abstract: Due to the small size of the rice seed images in the rice seedling tray and hole tray, perspective distortion occurs, and the size of each target in the image varies. Traditional convolutional neural networks using receptive field convolution kernels of the same size cannot accurately capture target features. Although some scholars have addressed this issue by combining density maps extracted from image blocks of different resolutions or feature maps obtained through convolutional filters of different sizes. By indiscriminately fusing information from all scales, these methods ignore the fact that the scales in the image are continuously changing. Training a classifier to predict the receptive field size used locally can solve this problem, but it is not an end-to-end training approach and cannot effectively address rapid scale changes. This article introduces multi-scale convolutional neural networks (MCNN), which explicitly extract features on multiple receptive field sizes and learn the importance of each such feature at each image position. This method adaptively encodes the scale of contextual information required to predict target density, thereby explaining potential rapid scale changes.

Keywords: Rice seedling tray; Rice seed counting; features; multi-scale convolutional neural networks.

1. Introduction

The main methods of transplanting rice seedlings include seedling throwing and transplanting. To achieve stable yield, the prerequisite is the seedling raising process. Especially in the machine transplanting method, the uniformity of seedling tray sowing is a prerequisite guarantee for the quality of seedling cultivation, indicating that seedling cultivation is the key factor restricting mechanical transplanting. For conventional rice seedling cultivation and transplanting, if the uniformity of seed distribution in the seedling tray is poor during rice seedling cultivation, it will lead to uneven seedling picking by the transplanter during the transplanting process; If there is a missed sowing in the seedling tray, it will cause a missed insertion phenomenon; If the proportion of voids is high, it will ultimately lead to a decrease in rice yield. For super rice, the number of seedlings used per hole is small, and the number of seeds sown is also small. Therefore, high sowing accuracy and low hole rate are required for mechanized seedling cultivation and sowing. Therefore, to ensure the quality of seedling cultivation, the first step is to test the density and quantity of rice seeds in the seedling tray.

Rice seed counting is an important research direction in the field of agricultural technology, aimed at improving harvesting efficiency and grain yield by accurately measuring the quantity of rice grains. At present, there are some research results on rice seed counting abroad. With the gradual maturity of computer vision and image processing technology, image recognition, morphological parameter measurement, particle counting, and other applications have been applied in crop breeding, quality identification, and other fields. At the same time, utilizing the powerful computing software MATLAB Matrix Lab, image processing such as image display and color space transformation can be achieved. In terms of grain counting, Jia Peng, Li Yongkui, and others used MATLAB grayscale processing, noise filtering, and binarization processing to achieve counting during the seed inspection process. Their results showed that the technical

accuracy of grains in the unpressed state could reach 100% [3]. Tian Mengxiang and others also used MATLAB to perform grayscale processing, image denoising, image segmentation, and other operations on rice grain images. In addition, distance transformation was performed to reduce the size of each grain using local minima, so as to separate the adhered grains and ensure the accuracy of grain counting. The final count of connected domains is the number of grains. Both single and sticky rice grains can be accurately and automatically counted, with an accuracy rate of up to 100% [4]. Ma Xu et al. [5] used convolutional neural networks (CNN) to detect the seeding amount of rice bowl floppy disks. They divided the bowl floppy disks into single hole images using grids, and then classified the number of rice seeds, simplifying the problem of rice seed quantity into a classification problem. Steven W. Chen et al. [6] (2017) used deep learning to extract candidate regions from the image of fruits using a half point detector with a fully convolutional network. They then estimated the number of fruits in each region using the algorithm of a second convolutional network. Finally, a linear regression model was used to count the total number of fruits, achieving the ability to count different types of fruits under uncontrollable light exposure and high occlusion.

This design uses the classic network model of deep learning for crowd counting tasks, Multi scale Convolutional Neural Networks (MCNN), to estimate the density of targets in input images, and the quantity can be obtained through integration. This method can better adapt to different objectives, enhance the applicability of the model, and at the same time, the density map can provide more information for operators. However, MCNN estimates the target density of each small unit in the image based on regression training to obtain the number of targets in the entire image, without directly detecting the number of targets in the entire image. For the scenario of using seedling trays, the density map can be used to discover whether the distribution of rice seeds is reasonable, which facilitates the full utilization of each

seedling tray while ensuring the normal development space of rice seeds.

2. Materials and Methods

2.1. Image acquisition and annotation

2.1.1. Dataset

Due to the original image resolution of 3648 pixels to 2736 pixels in the dataset, even if the original image is cut to a size suitable for annotation data, the resolution is still 608 pixels to 684 pixels. The size of the Ground_Truth and H5 files generated during training is tied to the resolution, and the image resolution needs to be reduced again. Due to the original image resolution of the entire seedling tray being 3648 pixels * 2736 pixels, the original image was cut into a size suitable for annotation data, with a resolution of 608

pixels * 684 pixels. The size of the Ground_Truth and H5 files generated during training is tied to the resolution, and the image resolution needs to be reduced again. The application scenario of this article is a rice seedling tray, which requires a self-made seedling tray dataset of 432 rice seeds (324 in the training set and 108 in the testing set), with an average resolution of 228 pixels * 256 pixels.

Use the tool CCLabeler to label the data and integrate it into a complete dataset. The overall size of the dataset is 198MB, and the file structure diagram is shown below. Among them, train_data and test_data are the training set and the testing set, respectively. The two folders store the rice seed images of the seedling tray in the format of .jpg and the .json file generated after marking the rice seeds. As shown in Figure 1.

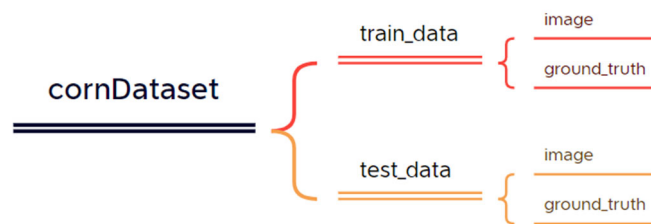


Figure 1. File structure diagram of the cornDataset dataset

2.1.2. Image annotation

The tool used for annotating rice seed data is CCLabeler, which is a Chinese text annotation tool designed to assist researchers and developers in annotating Chinese text. This tool is based on the Python language and utilizes various open-source toolkits and libraries, such as NLTK, Jieba, and Pandas. CCLabeler supports various types of annotation tasks, such as part of speech tagging, named entity recognition, sentiment analysis, etc. This tool aims to provide users with

an efficient text annotation experience and has a certain degree of flexibility and scalability to meet various needs. CCLabeler is particularly suitable for research and development work on annotating Chinese text.

Navigate to the CCLabeler directory in the cmd window, execute python manageability. py runserver 8000, enter localhost: 8000 in the browser address bar to enter the login interface, and enter the account information configured in the json folder to annotate the image.

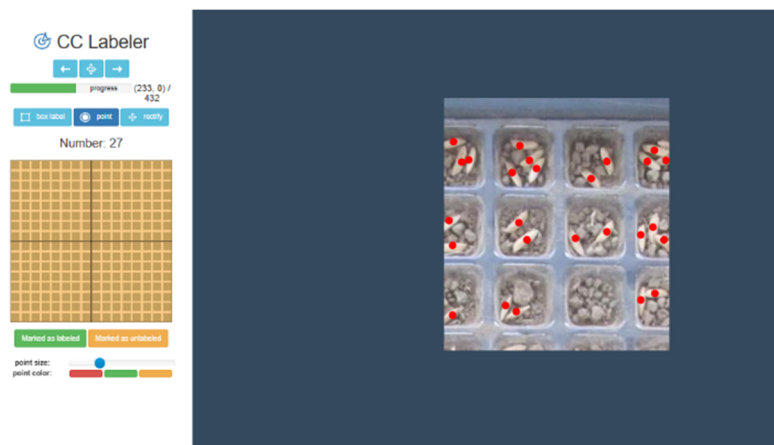


Figure 2. Image annotation operation interface

2.2. MCNN structural model

MCNN is a deep learning model based on multi-channel convolutional neural networks, specifically designed for object detection. Its main feature is the ability to extract various features such as the shape, texture, and color of the target, and effectively fuse these features. It improves the

accuracy and efficiency of the model through parallel computing of multiple convolutional layers. At the same time, it can perform multi-scale object detection at different resolutions, thereby identifying targets of different sizes and shapes.

Compared to traditional convolutional neural networks, MCNN adds multiple independent convolutional layers to the

network structure and combines them into a whole. MCNN typically consists of three main parts: input layer, multi column convolutional layer, and output layer. The input layer receives the image data to be processed, while the multi column convolutional layer is composed of multiple independent convolutional neural networks, each of which extracts features through different filters. These independent convolutional neural networks can simultaneously process different features, reducing the model's dependence on pooling layers while also increasing the model's robustness and generalization ability.

In Figure 3, it can be seen that MCNN consists of three parallel convolutional neural networks, using receptive fields of 9×9 , 7×7 , and 5×5 , and convolution kernels of 16, 20, and 24, respectively, to convolve the rice seed images in the seedling tray. At the same time, the step size of each convolutional layer is one pixel to ensure that the output size of each convolutional layer remains unchanged. Maximum pooling is used for each 2×2 region, and ReLU is used as the activation function. Using smaller convolution kernels for

larger convolutional neural networks makes the features learned by the network adaptive. MCNN replaces fully connected layers with 1×1 matrices to reduce model parameters: Fully connected layers typically introduce a large number of model parameters, increasing the complexity of the model, while 1×1 convolution kernels can reduce the number of channels without affecting the size of the feature map, thereby reducing the number of model parameters. Improving model efficiency: As a 1×1 convolution kernel can reduce the number of channels, it can reduce computational complexity and improve model efficiency, especially in situations where hardware resources such as GPUs are limited. Enhance feature expression ability: A 1×1 convolution kernel can linearly combine features from different channels to generate new feature representations, which can enhance the feature expression ability of the model. Avoiding overfitting: Using a 1×1 convolution kernel can increase the nonlinearity of the network, thereby improving the model's generalization ability and avoiding overfitting.

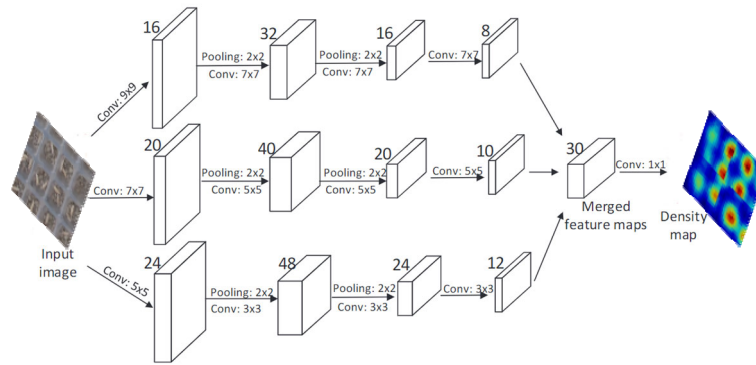


Figure 3. MCNN Network Model Structure

In the MCNN model, pooling operations and Spatial Pyramid Pooling strategies are adopted. In pooling operations, by reducing the size of feature maps, the parameters and computational complexity of the model can be reduced, while also preventing overfitting to a certain extent. In spatial pyramid pooling, the feature map size can be unified, multi-scale features can be extracted, the receptive field of the network can be increased, and the occurrence of overfitting can be reduced.

The structure of spatial pyramid pooling is as follows:

Input feature map: The input of SPP is a feature map, and the size of the feature map can be any size.

Block: Divide the input feature map into blocks and perform pooling operations on each block separately. The size and number of blocks are determined based on the design parameters of SPP and can include pooling layers of different scales to extract multi-scale features.

Pooling: Pooling each block and compressing it into a fixed size vector. Common pooling methods include max pooling, average pooling, etc.

Splicing: Splicing all pooled vectors together to form a feature vector of length N , where N is the total dimension of all block pooled vectors.

Output: Input the feature vectors into the fully connected layer for subsequent tasks such as classification.

Different sizes of input images may result in feature maps of different sizes, which can pose difficulties for subsequent

fully connected layers. By using SPP, the size of the feature map can be fixed to a fixed tensor, thereby unifying the size of the feature map and reducing the computational complexity of the fully connected layer. SPP can divide the input image into multiple regions, perform pooling operations on each region, and obtain feature representations within each region. By using pooling layers of different sizes in SPP, features of different scales can be extracted, thereby enhancing the network's ability to understand images. By using larger pooling layers in SPP, the receptive field of the network can be increased, enabling the network to capture broader contextual information and improve its ability to understand images. SPP can reduce the occurrence of overfitting without increasing network parameters. By pooling each region, the amount of data can be increased, thereby reducing the risk of overfitting.

3. Experimental Results and Analysis

3.1. Model parameter adjustment

PP PaddlePaddle VisualDL is used for visual parameter adjustment when training models. PP PaddlePaddle VisualDL is a visual tool for in-depth learning development, which aims to help developers better understand, optimize and debug their models. This tool provides many powerful features in visualization, including drawing loss, accuracy, scalars, and histograms. In addition, PP PaddlePaddle VisualDL also

supports visual model structure and gradient flow graph to monitor parameter changes and data flow during training. This tool not only makes the training process easier to understand, but also helps developers identify problems and bottlenecks in the model and optimize its performance. PP PaddlePaddle VisualDL is an open source tool, which can be downloaded and used through GitHub and the official website.

Observing the convergence status and minimum MAE of MAE with learning rates of $1e-2$, $1e-3$, and $1e-4$ at epoch 20, MAE refers to the Mean Absolute Error, which is an indicator for measuring the error of a prediction model. In statistics and machine learning, MAE is commonly used to evaluate the prediction error of a model on data points. The calculation method of MAE is to take the absolute difference between the predicted value and the actual value of each data point, and then take the average of all absolute differences, which is MAE. Compared with other error indicators, MAE has good interpretability and is easy to understand, and is insensitive to the influence of outliers. Meanwhile, MAE can also be used to compare the predictive performance of different models on

the same dataset. RMSE stands for Root Mean Square Error, which is a common metric used to measure the error of prediction models. In the fields of statistics and machine learning, RMSE is commonly used to measure the difference between model predictions and actual values. The calculation method of RMSE is to square the difference between the predicted value and the actual value of each data point, then take the average of all squared differences, and finally square the average to obtain RMSE. Compared with other error indicators, RMSE has a penalty power on the square of the error, which can better reflect the size and differences of prediction errors. RMSE can also be used to compare the prediction performance of different models on the same dataset, in order to determine the optimal prediction model. As shown in Figure 4 (RMSE is dark blue, MAE is light blue), it is observed that the lower limit of MAE is lower when the learning rate is $1e-4$, and MAE converges to 3.666 when the epoch is 200. Therefore, the best model for training is selected with a learning rate of $1e-4$ and an epoch of 200. The model was tested and found to have MAE=3.879 and RMSE=5.215.

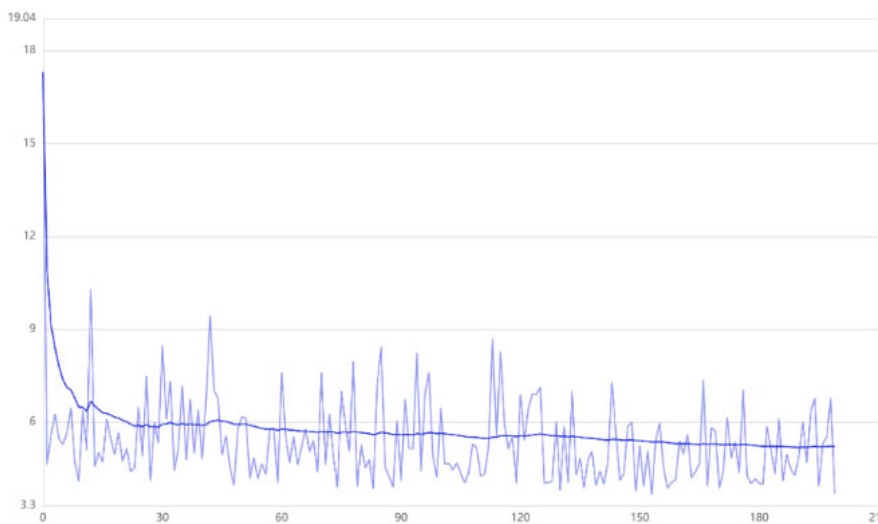


Figure 4. Results of MAE and RMSE Training

3.2. Model Testing and Analysis

Randomly select 20 rice seed images from the seedling tray for manual counting, and use the model trained in the

previous section to calculate the number of rice seeds. The density map of rice seeds on a bowl shaped floppy disk based on MCNN is shown in Figure 5.

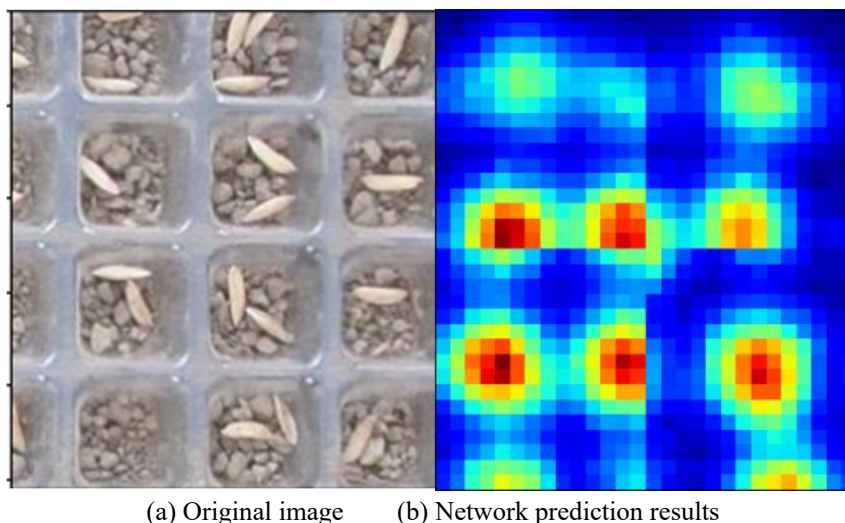


Figure 5. Density map of rice seeds on bowl shaped floppy disks based on MCNN

After randomly selecting 20 images for manual counting and network modeling, the statistical MAE of rice seed counting was only 0.065, which performed well. However, the vast majority of groups could not achieve 100%, and could basically complete the low requirement scenario of estimating and counting rice seed density in seedling trays.

Table 1. Three Scheme comparing

Number	Manual counting	Network prediction	Absolute error
1	23	24	0.043
2	18	17	0.055
3	34	32	0.058
4	22	19	0.136
5	20	19	0.050
6	39	41	0.051
7	50	46	0.080
8	45	47	0.044
9	39	38	0.025
10	47	48	0.021
11	49	45	0.081
12	38	37	0.026
13	44	46	0.045
14	45	46	0.022
15	46	46	0
16	37	43	0.162
17	42	42	0
18	18	16	0.222
19	21	16	0.238
20	19	20	0.052
MAE	-	-	0.065

4. Conclusion

This article is based on MCNN to detect the density and quantity of rice seeds in the seedling tray. The accuracy and efficiency of the model are improved through parallel computing of three columns of convolutional networks. The three networks respectively sense the features of different receptive fields, and even if the image resolution or shooting angle leads to different sizes of rice seeds, each column of convolutional network can still adapt. Create a dataset of rice seedling tray original images and design Python scripts for

automated processing to improve the efficiency of dataset processing. Mark the rice seeds in the graph using CCLabeler, generate an H5 file, and organize it into a usable dataset called cornDataset. Then start training the model. In the process of finding the best model parameters, use PP PaddlePaddle VisualDL to visually adjust the parameters. When adjusting the learning rate, it is found that the learning rate is the best at $1e-4$. Not only does MAE converge, but also the lower limit of MAE is the lowest. Therefore, choose epoch and learning rate parameters to train the best model. By comparing the statistics of network input images with manual counting, the accuracy of the model was analyzed. After comparing the results with manual counting, the average absolute error was only 0.065. This model can be applied to application scenarios with lower error requirements. The model still needs continuous improvement to enhance its accuracy.

Acknowledgment

College Student Innovation and Entrepreneurship Program Project (24C159).

References

- [1] Yuan Peichao, Ji Yao, Zhang Wenyi, et al. Research status and prospects of mechanized Transplanting of bowl-Seedlings in rice [J]. Journal of Chinese Agricultural Mechanization, 2025, 46 (04): 29-34+57.
- [2] Wang Zhicheng, Zhan Xiaokang, Xue Yang, et al. Effect of Piriformospora indica on Seedling Quality in Rice Mechanical Transplanting [J]. Journal of Agricultural Science and Technology, 1-12.
- [3] Jia Peng, Li Yongkui, Zhao Ping. Grain Counting Method Based on Matlab Image Processing [J]. Journal of Agricultural Mechanization Research, 2009, 31 (1): 152-153, 156.
- [4] Tian Mengxiang, Zhang Shilong, He Youxun, et al. A fast and efficient automatic counting method for rice grains [J]. Jiangsu Agricultural Science, 2014, 42 (2): 64-66
- [5] Deng Xiangwu, Ma Xu, Qi long, et al. Detection on rice seeds quantity per hole of the pot body seed tray based on convolutional network [J]. Journal of Chinese Agricultural Mechanization, 2020, 41(7): 130-136.
- [6] Steven W. Chen, Shreyas S. Shivakumar, Sandeep Dcunha, et al. Counting Apples and Orange With Deep Learning: A Data-Driven Approach[J]. IEEE Robotics and Automation Letters, 2017, 02(02): 781.