Insects identification with convolutional neural network technique in the sweet corn field

A P Naufal¹, C Kanjanaphachoat², A Wijaya³, N A Setiawan³ and R E Masithoh*¹

¹Department of Agricultural and Biosystems Engineering, Faculty of Agricultural Technology, Universitas Gadjah Mada, Yogyakarta, Indonesia
²Division of Agricultural Engineering, Faculty of Engineering and Agro-Industry, Maejo University, Chiang Mai, Thailand
³Department of Electrical Engineering and Information Technology, Faculty of Engineering, Universitas Gadjah Mada, Indonesia

Corresponding author: evi@ugm.ac.id

Abstract. A method to identify the type of insects with accurate and precise results is of importance. Nowadays, an automatic object identification system with increased accuracy, improved speed, and less cost have been developed. Convolutional Neural Network (CNN) implementation for image identification or classification can be done by collecting large-scale datasets containing hundreds to millions of images to study the many parameters involved in the network. This research was conducted to develop and apply the CNN model to identify eight species of insects in the sweet corn field in Thailand. Those insects were Calomycterus sp., Rhopalosiphum maidis, Frankliniella williamsi, Spodoptera frugiperda, Spodoptera litura, Ostrinia furnacalis, Mythimna separata, and Helicoverpa armigera. The CNN model in this research was built with four convolutional layers, which consist of Conv2D, batch normalization, max pooling, dropout sublayer, and a fully-connected layer. In total, 5568 images were trained with 10 trials and different train attempts for each trial, were then tested with 40 images. The result shows that the CNN model has succeeded in identifying images of sweet corn insects with 80% up to 95% prediction accuracy for images with no background.

1. Introduction

There are many challenges in agriculture, one of them is how to deal with crop pests. Crop pests, e.g. paddy pests, corn pests, soybean pests, etc., can damage the crop and affect the productivity of crop yield. It is difficult to classify insects due to the complex structure and high degree of similarity of the appearance between species [1]. It is important to identify insects in the crops at early stages to prevent the spread of insects and to select effective pesticides and biological control methods.

Image recognition for automatic identification is one of the popular methods compared to manual methods. An automatic identification system with increased accuracy, improved speed, and low cost is needed. Nowadays, the computer vision system (CVS) is still commonly used in image recognition or classification. The CVS consists of two main methods, i.e. image processing and deep learning for decision making. The image processing method includes image pre-processing, feature extraction, feature selection, and image understanding.

Deep learning is a type of machine learning that uses multilevel neural networks that allow computers to learn and extract deep abstract features automatically [2]. Deep learning allows computational models
that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction [3]. This method has improved the state-of-the-art in voice and visual object recognition, object detection, and many other domains. Deep learning is a popular method in the image recognition field [4] with wide applications from detecting fruits [5], color additives in food [6], and other food applications [7]. This technique will increase the effectiveness of object identification but requires more image processing technique. However, the problem can be solved by using a convolutional neural network (CNN) technique [8].

CNN is a type of artificial neural network used in image processing that is specifically designed to process pixel data. CNN can automatically perform both for feature extraction and selection under the network’s depth and weight sharing between nodes which can help alleviate over-fitting [9]. The implementation of CNN for image identification or classification can be done by collecting such a large-scale dataset that contains hundreds to millions of images for network training because of the need for learning many parameters involved in the network. This situation can be simplified by applying the CNN model that has been pre-trained based on large-scale image data which is called transfer learning [10]. Features learned by deep CNN have been recognized to be more robust and expressive than handcrafted ones [11].

CNN consists of four types of layers: input layer, convolutional layer, pooling layer, and fully connected layer. Those multilayer architectures of the artificial neural network were developed by LeNet-5, ZFNet, VGGNet, GoogLeNet, and ResNet [12]. When these layers are stacked, a CNN architecture has been formed. The input layer will hold the pixel values of the image. The convolutional layer focuses on the use of learnable kernels; these are usually small in spatial dimensionality but spreads along the entirety of the depth of the input. Convolution layers in a CNN play the important role of feature extractor [13]. When the data hits a convolutional layer, the layer convolves each filter across the spatial dimensionality of the input to produce a 2D activation map. The convolution operation extracts different features of the input. The first convolution layer extracts low-level features (e.g. edges, lines, and corners), then the higher-level layers extract higher-level features [13]. The pooling layer is supposed to reduce the dimensionality of the representation gradually and reduce the number of parameters and the computational complexity of the model. The fully connected layer contains neurons of which are directly connected to the neurons in the two adjacent layers, without being connected to any layers within them [14].

CNN has been successfully used in different computer vision tasks such as object detection, pattern recognition, and image understanding. In this research, the CNN model is used to provide high accuracy in insect classification. To do so, pre-trained CNN models using transfer learning are applied for comparison of classification accuracy. The developed CNN model will be used to recognize the different types of insects in crop fields at an early stage to improve crop quality and productivity. The objectives of this research are to build a CNN model to identify the type of insects in the sweet corn field and to evaluate the performance of the developed CNN model in identifying insects.

2. Materials and methods

2.1. Materials and general method
Materials used in this research are images of Calomycterus sp., Rhopalosiphum maidis, Frankliniella williamsi, Spodoptera frugiperda, Spodoptera litura, Ostrinia furnacalis, Mythimna separata, and Helicoverpa armigera insects. Tools used in this research is personal computer installed with Windows 10 Education 64-bit operating system, Google Colaboratory as Python online notebook, Python programming language, Keras library for deep neural network purposes, TensorFlow library as a backend for generating machine learning models, NumPy library to perform complex scientific computations or calculations, Matplotlib library to present data that has been processed in two-dimensional graphs, and Pandas library for data manipulation and analysis purposes. Mozilla Firefox as a web browser to open Google Colab notebook and retrieve images, and Adobe Photoshop as an image
editing software to preprocess the images manually were also used. In general, the flowchart of this research is shown in Figure 1.

![Research flowchart.](image)

**Figure 1.** Research flowchart.

2.2. The datasets gathering and image pre-processing

The first step of this research is collecting the images of eight sweet corn insects. The image of each type of insects is retrieved from the web as much as possible. This huge number of images called the dataset. The list of datasets in this research is shown in Table 1. These images then processed using Adobe Photoshop to maintain the brightness and contrast, rotate the image, cut the background or adjust the size.

| Class num | Class name (label) | Type of insect | Num of image |
|-----------|--------------------|----------------|--------------|
| 1         | aphids             | *Calomycterus* sp. (Corn Soil Moth) | 560          |
| 2         | armigera           | *Rhopalosiphum maidis* (Corn Aphids) | 816          |
| 3         | calomycterus       | *Frankliniella williamsi* (Corn Thrips) | 648          |
| 4         | faw                | *Spodoptera frugiperda* (Fall Armyworm) | 1,240        |
| 5         | furnacalis         | *Spodoptera litura* (Vegetable Worm) | 480          |
| 6         | litura             | *Ostrinia furnacalis* (Corn Stalk Borer) | 480          |
| 7         | separata           | *Mythimna separata* (Cornworm) | 488          |
| 8         | thrips             | *Helicoverpa armigera* (Cotton Anchor Borer) | 856          |
| **Total image** | | | **5,568** |

2.3. CNN model development, training, and testing

The CNN model was built with a Python programming language on the Google Colaboratory notebook. Some Python libraries are imported and used to handle this CNN task, such as Keras, NumPy, Matplotlib, and Pandas. The architecture of the CNN model built in this research is shown in Figure 2.
2.3. Dataset and model training

The dataset then imported from the CNN model to be loaded and trained. This training stage uses 50 epochs (the number of times where the model trains these objects). The CNN model is then used to predict desired images. The result is in the form of character value inside the brackets “(...)” located below the tested pictures and next to the file name of the pictures.

2.4. Prediction accuracy evaluation

The CNN model is shown as a two-dimensional graph. The graph consists of accuracy and loss level between the training and validation process with the number of epochs. The process of prediction uses a total of 40 test images, which consist of 5 test images for each class or category. A basic mathematical equation is used to calculate the real prediction accuracy, as shown in Equation 1.

\[
\text{Prediction accuracy} (%) = \frac{\text{correct predicted image}}{\text{total test image}} \times 100
\]  

The result of predicted images is then arranged into a confusion matrix for each prediction and trials. The confusion matrix consists of 8 actual insects as a row or y-axis, and 8 predicted insects as a column or x-axis. The scenario-based evaluation is then applied to analyze the result of prediction deeply. This scenario was created by ordering kinds of insects with similar physical characteristics into one group. The first scenario is an original result of prediction without any changes. The second scenario is a prediction result with changes in merged “faw” and “litura” categories into one “faw/lit” category, also “separata” and “armigera” categories into one “sep/arm” category. This grouping is done since the “faw” and “litura” categories also “separata” and “armigera” categories have the most similarity in a physical characteristic. The third scenario is a prediction result with changes in merged “separata”, “armigera”, and “furnacalis” categories into one “sep/arm/fur” category, but the “faw/lit” category which merged before in scenario 2 remain. This grouping is done since the “furnacalis” category looks similar to “separata” and “armigera” categories, although not as close as similarity of “separata” and “armigera” categories. The fourth scenario is a prediction result with changes in merged “faw”, “litura”, “separata”, “armigera”, and “furnacalis” categories into one “fa/li/se/ar/fu” category. This grouping is done since these categories have similar physical characteristics in general, although in detail these are not. The comparison is shown in Figure 3 for “faw”, “litura”, “separata”, “armigera”, and “furnacalis” category.

(a) | (b) | (c) | (d) | (e)  
---|---|---|---|---
Figure 3. Comparison between “faw” (a), “litura” (b), “separata” (c), “armigera” (d), and “furnacalis” (e) category.
3. Results and discussion
The prediction in this research uses test images with no background, or the background color is pure white. There are 40 test images, with 5 images for each insect category. The predicted insects are visualized by the model, as shown in Figure 4.

![Figure 4. Visualization of insects prediction results.](image)

The result of predicted images is then arranged into a confusion matrix for each trial. In this paper, the confusion matrix is shown for only the highest and lowest value of prediction accuracy, as shown in Table 2.

Table 2. Confusion matrix of the highest (fifth trial) and lowest (second trial) prediction accuracy.

| Actual insects | Predicted insects of the fifth trial (highest prediction accuracy) | Predicted insects of the second trial (lowest prediction accuracy) |
|----------------|---------------------------------------------------------------|---------------------------------------------------------------|
|                | Aph   | Arm   | Cal   | Faw   | Fur   | Lit   | Sep   | Thr   | Aph   | Arm   | Cal   | Faw   | Fur   | Lit   | Sep   | Thr   |
| Aph            | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 5     | 0     | 0     | 0     | 0     | 0     | 0     | 0     |
| Arm            | 0     | 5     | 0     | 1     | 0     | 3     | 0     | 0     | 0     | 5     | 0     | 0     | 1     | 0     | 3     | 0     |
| Cal            | 0     | 0     | 3     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 3     | 0     | 0     | 0     | 0     | 0     |
| Faw            | 0     | 0     | 2     | 4     | 0     | 0     | 0     | 0     | 0     | 0     | 2     | 4     | 0     | 0     | 0     | 0     |
| Fur            | 0     | 0     | 0     | 0     | 3     | 0     | 0     | 0     | 0     | 0     | 0     | 3     | 0     | 0     | 0     | 0     |
| Lit            | 0     | 0     | 0     | 1     | 1     | 5     | 0     | 0     | 0     | 0     | 0     | 1     | 1     | 5     | 0     | 0     |
| Sep            | 0     | 0     | 0     | 0     | 0     | 2     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 2     | 0     |
| Thr            | 0     | 0     | 0     | 0     | 0     | 0     | 5     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 0     | 5     |

Table 3. Prediction accuracy in any scenario.

| No. (Trial) | Train Accuracy (%) | Number of Train (Attempt) | Prediction Accuracy (%) |
|-------------|--------------------|---------------------------|-------------------------|
|             | Scenario 1         | Scenario 2                | Scenario 3               | Scenario 4               |
| 1           | 94.31              | 77.50                     | 80.00                   | 85.00                   | 87.50               |
| 2           | 78.85              | 12.50                     | 27.50                   | 30.00                   | 57.50               |
| 3           | 86.39              | 67.50                     | 80.00                   | 82.50                   | 85.00               |
| 4           | 88.57              | 72.50                     | 87.50                   | 92.50                   | 95.00               |
| 5           | 90.01              | 80.00                     | 90.00                   | 92.50                   | 95.00               |
| 6           | 87.28              | 75.00                     | 85.00                   | 95.00                   | 95.00               |
| 7           | 87.08              | 65.00                     | 72.50                   | 82.50                   | 87.50               |
| 8           | 90.28              | 72.50                     | 87.50                   | 90.00                   | 95.00               |
| 9           | 86.90              | 22.50                     | 30.00                   | 35.00                   | 40.00               |
| 10          | 87.28              | 80.00                     | 87.50                   | 87.50                   | 87.50               |

The confusion matrix above shows that among the 5 test images for each type or category of insects, the CNN model in the fifth trial has been successfully identifying 2 images of separata insect, 3 images of calomycterus and furnacalis insects, 4 images of faw insect, and 5 images of aphids, armigera, litura, and thrips insects. There is no zero value in this result, which means all of the insect categories are correctly identified by the model. Meanwhile, the CNN model in the second trial has been successfully identifying 1 image of furnacalis insect, and 4 images of litura insect. However, in this second trial, the model failed to identify the aphids, armigera, calomycterus, faw, separata, and thrips insects, as there
are 0 insects identified correctly. The comparison of train accuracy and prediction accuracy in any scenario for each trial can be seen in Table 3.

In addition, the graph as in Figure 5 is generated to clearly explain and evaluate prediction accuracy with train accuracy and the scenarios. All initial (original) prediction accuracy, which is a scenario 1, mostly has a value greater than 60%. From a total of 40 test images, up to 32 images, or 80% of images were correctly predicted by the model. It happened since the test images have no background, or the background is white. The model mostly can predict the images correctly since the model was trained with a background-less (white background) train and validation images. However, the model unable to predict some images correctly. Those wrong prediction can be found in the second and ninth trial, which got only 5 and 9 images or 12.5% and 22.5% of images were correctly predicted, respectively.

The higher value of train accuracy does not mean a higher value of real accuracy, which is prediction accuracy. This statement is the same as the first prediction. The first trial with the highest value of train accuracy and trained 4 times got 77.5% prediction accuracy, but the tenth trial which has the train accuracy value smaller than the first trial got 80% prediction accuracy which is higher than the first trial. The second trial with the fewest value of train accuracy and trained only 1 time got only 12.5% prediction accuracy, which makes sense at all. However, the ninth trial with 86.9% train accuracy and trained 2 times got only 22.5% prediction accuracy. The value of train accuracy, in this ninth trial case, is high enough, yet the prediction accuracy value is very small.

The greater number of train process does not mean the higher value of train accuracy and prediction accuracy. That makes sense if the second trial got only 12.5% of prediction accuracy since the model trained only 1 time and the value of train accuracy is not high enough. The third, seventh, and ninth trial have the same number of trains, also the value of train accuracy are not quite different. However, the ninth trial got only 22.5% of prediction accuracy, while the third and seventh trial got 67.5% and 65% of prediction accuracy, which means they have prediction accuracy value higher than the ninth trial with the same number of trains. The eighth trial with the highest number of trains, 12 times, has 90.28% of train accuracy value which is high enough, but the value of prediction accuracy is not the highest one, which is 72.5%. This case can be compared to the highest value of prediction accuracy, there are the fifth and tenth trials, which have 80% prediction accuracy value, but the fifth trial took 5 times of training, moreover, the tenth trial took only 3 times of training.

The scenario-based evaluation is applied in this prediction result to evaluate the result of prediction accuracy. The first scenario got the highest value of prediction accuracy on the fifth and tenth trial with 80% of prediction accuracy, and the lowest value on the second trial with only 12.5% of prediction accuracy. The second scenario got the highest value of prediction accuracy on the fifth trial with 90% prediction accuracy, and the lowest value still on the second trial with only 27.5% prediction accuracy. The third scenario got the highest value of prediction accuracy on the sixth trial with 95% prediction accuracy, and the lowest value still on the second trial with only 30% prediction accuracy. The last
scenario, scenario 4, got the highest value of prediction accuracy on the fourth, fifth, sixth, and eighth trial with 95% of prediction accuracy, and the lowest value on the ninth trial with only 40% of prediction accuracy.

To understand which trial, train accuracy value, and a number of the train gave the best value of the prediction accuracy in each scenario, the highest value of prediction accuracy is selected and arranged, as shown in Table 4 below.

**Table 4. Prediction accuracy for any scenario.**

| Scenario | No. (Trial) | Train Accuracy (%) | Num of Train (Attempt) | Prediction Accuracy (%) |
|----------|-------------|--------------------|------------------------|-------------------------|
| 1        | 5           | 90.01              | 5                      | 80.00                   |
|          | 10          | 87.28              | 3                      | 80.00                   |
| 2        | 5           | 90.01              | 5                      | 90.00                   |
| 3        | 6           | 87.28              | 3                      | 95.00                   |
|          | 4           | 88.57              | 3                      | 95.00                   |
|          | 5           | 90.01              | 5                      | 95.00                   |
|          | 6           | 87.28              | 3                      | 95.00                   |
|          | 8           | 90.28              | 12                     | 95.00                   |

Table 4 shows the highest value of prediction accuracy for each scenario. In scenario 1, the highest value of prediction accuracy is 80% with 90.01% of train accuracy and 5 times of train attempt, also 87.28% of train accuracy and 3 times of train attempt. In scenario 2, the highest value of prediction accuracy is 90% with 90.01% of train accuracy and 5 times of train attempt, the same as scenario 1. In scenario 3, the highest value of prediction accuracy is 95% with 87.28% of train accuracy and 3 times of train attempt. In scenario 4, the highest value of prediction accuracy is 95% with 88.57% of train accuracy and 3 times of train attempt, 90.01% of train accuracy and 5 times of train attempt (the same as scenario 1 and 2), 87.28% of train accuracy and 3 times of train attempt (the same as scenario 3), also 90.28% of train accuracy and 12 times of train attempt. The model can predict the images of insects with the prediction accuracy value of 80% up to 95%.

4. Conclusion
The model of Convolutional Neural Network (CNN) in this research was built with 4 convolutional layers, which consist of Conv2D, batch normalization, max pooling, dropout sublayer, and a fully-connected layer. The developed CNN model has succeeded in identifying images of sweet corn insects with 80% up to 95% prediction accuracy for images with no background.

Acknowledgment
This research was funded by the Department of Agricultural and Biosystems Engineering Universitas Gadjah Mada, Indonesia, and Division of Agricultural Engineering Maejo University, Thailand, under the student exchange program 2019.

References
[1] Thenmozhi K and Srinivasulu R U 2019 *Comput. Electron. Agric.* 164 104906
[2] Li Y, Wang H, Dang L M, Sadeghi-Niaraki A and Moon H 2020 *Comput. Electron. Agric.* 169 105174
[3] LeCun Y, Bengio Y and Hinton G 2015 *Nature* 521 436–444
[4] Zhang W, Zhao D, Gong W, Li Z, Lu Q and Yang S 2016 *Proceedings - 2015 IEEE 12th International Conference on Ubiquitous Intelligence and Computing* (United States) pp 690–693
[5] Fu L, Feng Y, Majeed Y, Zhang X, Zhang J, Karkee M and Zhang Q 2018 *IFAC-PapersOnLine* 51 45–50
[6] Pribadi W, Masithoh R E, Nugroho A P and Radi 2019 *IOP Conf. Ser. Earth Environ. Sci.* **355** 012003

[7] Chen H, Xu J, Xiao G, Wu Q and Zhang S 2018 *J. Parallel Distrib. Comput.* **117** 218–227

[8] Thenmozhi K, Srinivasulu, Reddy U, Termritthikun C, Muneesawang P and Kanprachar S 2019 *J. Telecommun. Electron. Comput. Eng.* **9** 63–67

[9] Nasiri A, Taheri-Garavand A and Zhang Y D 2019 *Postharvest Biol. Technol.* **153** 133–141

[10] Lu Y 2016 *arXiv* 1612.00983

[11] Ciocca G, Napoletano P and Schettini R 2018 *Comput. Vis. Image Underst.* **176–177** 70–77

[12] Gu J, Wang Z, Kuen J, Ma J, Shahroudy A, Shuai B, Liu T, Wang X, Wang G, Cai J and Chen T 2018 *Pattern Recognit.* **77** 354–377

[13] Hijazi S, Kumar R and Rowen C 2015 *Using Convolutional Neural Networks for Image Recognition* (California, USA: Cadence)

[14] O’Shea K and Nash R 2015 *arXiv* 1511.08458 1–11