**University of
Zurich**^{UZH}

**Zurich Open Repository and
Archive**

University of Zurich
University Library
Strickhofstrasse 39
CH-8057 Zurich
www.zora.uzh.ch

Year: 2021

Using a two-stage convolutional neural network to rapidly identify tiny herbivorous beetles in the field

Takimoto, Hironori ; Sato, Yasuhiro ; Nagano, Atsushi J ; Shimizu, Kentaro K ; Kanagawa, Akihiro

Abstract: Recently, deep convolutional neural networks (CNN) have been adopted to help non-experts identify insect species from field images. However, the application of these methods on the rapid identification of tiny congeneric species moving across heterogeneous background remains difficult. To improve rapid and automatic identification in the field, we customized an existing CNN-based method for a field video involving two *Phyllotreta* beetles. We first performed data augmentation using transformations, syntheses, and random erasing of the original images. We then proposed a two-stage method for the detection and identification of small insects based on CNN, where YOLOv4 and EfficientNet were used as a detector and a classifier, respectively. Evaluation of the model revealed that one-step object detection by YOLOv4 alone was not precise (Precision = 0.55) when classifying two species of flea beetles and background objects. In contrast, the two-step CNNs improved the precision (Precision = 0.89) with moderate accuracy (F-measure = 0.55) and acceptable speed (ca. 5 frames per second for full HD images) of detection and identification of insect species in the field. Although real-time identification of tiny insects remains a challenge in the field, our method aids in improving small object detection on a heterogeneous background.

DOI: <https://doi.org/10.1016/j.ecoinf.2021.101466>

Posted at the Zurich Open Repository and Archive, University of Zurich

ZORA URL: <https://doi.org/10.5167/uzh-208494>

Journal Article

Accepted Version

The following work is licensed under a Creative Commons: Attribution-NonCommercial-NoDerivatives 4.0 International (CC BY-NC-ND 4.0) License.

Originally published at:

Takimoto, Hironori; Sato, Yasuhiro; Nagano, Atsushi J; Shimizu, Kentaro K; Kanagawa, Akihiro (2021). Using a two-stage convolutional neural network to rapidly identify tiny herbivorous beetles in the field. *Ecological Informatics*, 66:101466.

DOI: <https://doi.org/10.1016/j.ecoinf.2021.101466>

Journal Pre-proof

Using a two-stage convolutional neural network to rapidly identify tiny herbivorous beetles in the field

Hironori Takimoto, Yasuhiro Sato, Atsushi J. Nagano, Kentaro K. Shimizu, Akihiro Kanagawa

PII: S1574-9541(21)00257-0

DOI: <https://doi.org/10.1016/j.ecoinf.2021.101466>

Reference: ECOINF 101466

To appear in: *Ecological Informatics*

Please cite this article as: H. Takimoto, Y. Sato, A.J. Nagano, et al., Using a two-stage convolutional neural network to rapidly identify tiny herbivorous beetles in the field, *Ecological Informatics* (2021), <https://doi.org/10.1016/j.ecoinf.2021.101466>

This is a PDF file of an article that has undergone enhancements after acceptance, such as the addition of a cover page and metadata, and formatting for readability, but it is not yet the definitive version of record. This version will undergo additional copyediting, typesetting and review before it is published in its final form, but we are providing this version to give early visibility of the article. Please note that, during the production process, errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.

© 2021 Published by Elsevier B.V.

Using a two-stage convolutional neural network to rapidly identify tiny herbivorous beetles in the field

Hironori Takimoto^{a,*,1}, Yasuhiro Sato^{b,c,1}, Atsushi J. Nagano^{d,e}, Kentaro K. Shimizu^{c,f} and Akihiro Kanagawa^a

^aFaculty of Computer Science and Systems Engineering, Okayama Prefectural University, Kuboki 111, Soja 719-1197, Okayama, Japan

^bResearch Institute for Food and Agriculture, Ryukoku University, Yokotani 1-5, Seta Oe-cho, Otsu 520-2194, Shiga, Japan

^cDepartment of Evolutionary Biology and Environmental Studies, University of Zurich, Winterthurerstrasse 190, 8057 Zurich, Switzerland

^dFaculty of Agriculture, Ryukoku University, Yokotani 1-5, Seta Oe-cho, Otsu 520-2194, Shiga, Japan

^eInstitute for Advanced Biosciences, Keio University, Nipponkoku 403-1, Daihouji, Tsuruoka 997-0017, Yamagata, Japan

^fKihara Institute for Biological Research, Yokohama City University, Maioka 641-12, Yokohama 244-08137, Kanagawa, Japan

ARTICLE INFO

Keywords:

entomology
fine-grained image classification
deep learning
herbivory
small object detection

ABSTRACT

Recently, deep convolutional neural networks (CNN) have been adopted to help non-experts identify insect species from field images. However, the application of these methods on the rapid identification of tiny congeneric species moving across heterogeneous background remains difficult. To improve rapid and automatic identification in the field, we customized an existing CNN-based method for a field video involving two *Phyllotreta* beetles. We first performed data augmentation using transformations, syntheses, and random erasing of the original images. We then proposed a two-stage method for the detection and identification of small insects based on CNN, where YOLOv4 and EfficientNet were used as a detector and a classifier, respectively. Evaluation of the model revealed that one-step object detection by YOLOv4 alone was not precise (Precision = 0.55) when classifying two species of flea beetles and background objects. In contrast, the two-step CNNs improved the precision (Precision = 0.89) with moderate accuracy (F-measure = 0.55) and acceptable speed (ca. 5 frames per second for full HD images) of detection and identification of insect species in the field. Although real-time identification of tiny insects remains a challenge in the field, our method aids in improving small object detection on a heterogeneous background.

1 Introduction

Insects are some of the most abundant groups of animals in the terrestrial ecosystem. Rapid and accurate identification of insect species is essential for pest management (Park et al., 2020; Preti et al., 2020) and species-level identification in biodiversity monitoring (Hogeweg et al., 2019; Høye et al., 2021). To augment the expertise in insect taxonomy and ecology, several attempts have been made to develop automated identification systems (Martineau et al., 2017). However, handcrafted extraction of image features requires experience on image processing (Mayo and Watson, 2007; Samanta and Ghosh, 2012; Yalcin, 2015; Xie et al., 2015; Rani and Amsini, 2016). Recently, the deep convolutional neural network (CNN) has revolutionized this task by facilitating the automatic extraction of key features from a huge number of insect images (Valan et al., 2019; Park et al., 2020; Almyrad and Kutucu, 2020; Hansen et al., 2020; Høye et al., 2021).

Despite its promising implications, the usage of CNN is largely restricted to a snapshot image taken on a simple background, such as a capturing sheet (Hansen et al., 2020) and trapping system (Preti et al., 2020). In contrast, field images are much more complex since they involve a

heterogeneous background with multiple species and individuals (Martineau et al., 2017). Furthermore, a snapshot or interval camera may not be sufficient to capture the key characteristics of species identification (Preti et al., 2020). A series of images, namely a video, may solve this issue by capturing multiple posing for an individual insect (Høye et al., 2021). Such video images are now easily taken using a handy device, such as a webcam or smartphone. Thus, a field video taken by a handy camera may provide a unique material for the convenient and thorough imaging of a target insect.

In the context of computer vision, CNN-based object detection has now enabled fast and accurate processing of a large series of images that involve living organisms and artificial objects (Liu et al., 2019). The goal of object detection is not only to determine whether objects from general categories are present on an image, but also to return the spatial location and determine the extent of each object by using a bounding box. A CNN is designed to automatically learn the spatial-feature hierarchies layer by layer and the consequent sub-sampling layers in a feature hierarchy that results in an inherent multiscale pyramid (Yamashita et al., 2018). It is well recognized that the higher layers are the most robust in terms of object pose, illumination, part deformation, and other variations because they have large receptive fields and strong semantics. However, a critical disadvantage in this step is the loss of geometric details and the low resolution, such that small objects more likely lack critical features since they are processed in subsequent layers (Liu et al., 2019).

*Corresponding author

✉ takimoto@c.oka-pu.ac.jp (H. Takimoto)

ORCID(s): 0000-0002-8795-7109 (H. Takimoto); 0000-0002-6466-723X (Y. Sato); 0000-0001-7891-5049 (A.J. Nagano); 0000-0002-6483-1781 (K.K. Shimizu)

¹H. Takimoto and Y. Sato contributed equally to this study.

Figure 1: Flow of the proposed two-stage detection method. During the first stage (left), the proposed method detected several candidate regions from an input image by YOLOv4 with a low threshold of detection. In the second stage (right), all candidate regions were re-identified using the EfficientNet.

Figure 2: Example of a captured image in a real field environment. Two species of *Phyllotreta* beetles (red square; ca. 2 mm in body length) simultaneously appeared on an individual *Arabidopsis thaliana*.

Thus, the detection of small objects using general-purpose CNNs is more difficult than that of large objects.

A large training dataset is required to generalize CNN models (LeCun et al., 2015). However, currently available insect images in public datasets may not be sufficient to train CNN models for real-world variations in object angles, camera conditions, and field background. To efficiently train CNN models for an individual insect, we need to take multi-angle photographs in a systematic manner (Hansen et al., 2020; Høye et al., 2021). However, in this method, the background is likely homogenous, which is not the case in fields with complex and heterogenous background. In contrast, citizen scientists may provide a large number of field photographs (Silvertown et al., 2015; Van Horn et al., 2018), but the database may be imbalanced with respect to species, angles, background objects, and camera devices. Synthesis or transformation of original images, called data augmentation (Shorten and Khoshgoftaar, 2019), is essential to create effective variation in training data with the cost to collect actual photographs kept minimum.

This study aimed to improve an existing method of object detection for a field video of small insects. Based on the idea of rapid and multiscale object detection (Van Etten, 2018), we proposed a two-stage detection and identification method for small insects (Fig. 1). Our proposed method aimed to address specific problems regarding the size, appearance, training data, and processing time of insect video in the field. For rapid detection of target insects, we focused on an object detection method called You Only Look Once (YOLO) v4 (Bochkovskiy et al., 2020). YOLOv4 is a single-shot object detector that is trained end-to-end from raw pixels to ultimate categories. When the size of a target object is relatively large, YOLOv4 can detect the object accurately (Ovchinnikova et al., 2021). Contrastingly, when the size of the target object is small, the object is often undetected or over-detected by YOLOv4. To address this issue, we first used YOLOv4 as a region proposal network, which detects many possible candidate regions in the entire system. All detected candidates were then re-identified by a CNN-based classifier, the EfficientNet (Tan and Le, 2019). With a less computational cost than other models, the EfficientNet retains sufficient capacity over various benchmark datasets by adopting a compound scaling method that enlarges the network depth, width, and resolution. In addition, we proposed a data augmentation method to train the two-stage CNNs as a generally applicable model under various camera or field conditions.

Material and methods

Study species and site

The target insects were the striped flea beetle *Phyllotreta striolata* and the turnip flea beetle *Phyllotreta atra*. The two insect species were also known as pests of *Brassica* crops (Ahuja et al., 2010). These insect species often appeared on the genus *Arabidopsis* (Brassicaceae) during the early summer days (Sato et al., 2019; Shimizu-Inatsugi et al., 2021). In this study, a total of 3800 potted individuals of *Arabidopsis thaliana* were transplanted in the field garden at the University of Zurich at the Irchel campus (Zurich, Switzerland; 47°06'N, 134°56'E) in July 2017 (1600 pots), July 2018 (1500 pots), and July 2019 (600 pots). Figure 2 shows a sample image of the two insect species appearing on an *A. thaliana* plant. Both species had a length of approximately 2 mm and had similar habitus. The only difference in their appearance was the stripes on their back. *P. striolata* had yellow stripes, while *P. atra* had none. Furthermore, both beetles created small holes when eating plant leaves, which might explain the appearance of gaps between the leaves (a lower blue square of Fig. 2) or a leaf hole in the backgrounds of images. The formation of these holes might be attributed to the feeding behavior of flea beetles (an upper blue square of Fig. 2).

Data collection

For training data, we first collected images of *P. striolata* and *P. atra* via Google search and through our fieldwork, and then selected foreground images for the data augmentation below. Our own photographs were taken during the transplant experiment in 2017 and 2018. We used a RICOH WG-4 digital camera (RICOH Company, Ltd., Tokyo, Japan), which generated 258 photographs (1,920×1,080 pixels) for *P. striolata* only, 106 photographs for *P. atra* only, and 41 photographs of both *P. striolata* and *P. atra*. For the Google search, we used the following keywords: “*Phyllotreta striolata*”, “*Phyllotreta atra*”, “striped flea beetle” (common name of *P. striolata* in English), “turnip flea beetle” (common name of *P. atra* in English), “kisuji-nomi-hamushi” (common name of *P. striolata* in Japanese, searched in Japanese katakana letters), or “mumon-kisuji-nomi-hamushi” (common name of *P. atra* in Japanese, searched in Japanese katakana letters). Hence, our search resulted in a total of 142 and 49 images for *P. striolata* and *P. atra*, respectively. The size of internet images ranged from 125×118 to 1427×939 pixel (461.4×342.8 pixel on average). However, due to the low resolution of the images and the high similarity of the two species, the effective number of high-resolution images with different posing for training data was only 32 and 29 for *P. striolata* and *P. atra*, respectively. Due to the copyright

of internet images, only the authors' original photographs were deposited on the GitHub repository (<https://github.com/h-taki/identification-of-tiny-insects>).

To procure a test video, we used a full HD video mode (1,920×1,080 pixels) of Logicoool C920r webcam (Logicoool Co. Ltd., Lausanne, Switzerland). A test video was photographed in 2019 to differentiate the survey years between the training and test data. The handy camera was approached to target insects as close as possible. To set a ground truth, we annotated the two species on 215 annotated video frames using the LabelImg tool (<https://github.com/tzutalin/labelImg>). The original video was taken in 30 FPS, but the nearest time frames are almost the same and costly for manual annotation; thus, every five frames were annotated. The series of annotated frames captured 640 insect objects, with 215 *P. striolata* and 425 *P. atra*. The annotated test video is available as Supplementary Video S1.

Data augmentation

Although optimizing many parameters of CNN based on deep learning required a large dataset, there were only small datasets for pest detection and identification. Existing datasets only had approximately 100 images for each type (Samanta and Ghosh, 2012; Rustia et al., 2018; Xia et al., 2018; He et al., 2019; Wang et al., 2012; Venugoban and Ramanan, 2014; Xie et al., 2018; Deng et al., 2018). To obtain a more generalized model, training data should have fine diversity as the objects varied in size, lighting conditions, and poses. To circumvent the limited quantity and diversity of training data, data augmentation was a useful technique that could increase the size of the training set without acquiring new images. Data augmentation was typically performed as part of dataset preprocessing for training an image detection and classification model, such as the CNN-based model (Shorten and Khoshgoftaar, 2019). The procedure included cropping, flipping, scaling, rotating, adding noise, and color transformation.

In this study, we used augmented datasets for the two-stage detection. First, we proposed data augmentation to effectively train YOLOv4 for the sake of a region proposal network in the entire system. Randomly segmented objects (i.e., segmented insect images) were pasted into natural images, such as photographs of soils and/or plants. Examples of synthesized images are shown in Fig. 3. A mask image as a foreground image was created by extracting only insect regions from the original images. A single image was then randomly selected from a large natural image dataset that did not contain any insects. A background image was cropped at 512×512 pixels from the selected image. The procedure of data augmentation consisted of conventional image processing methods, image synthesis (Fig. 3c), and random erasing (Fig. 3a, b). First, the foreground and background images were subjected to image processing operations such as cropping, flipping, and scaling. By combining a foreground image with a background image, we constructed a large training image dataset. Overlapping foreground images were allowed when combining multiple foreground images into

Table 1

The parameters of the data augmentation for training YOLOv4 (a) or EfficientNet (b).

(a) For YOLOv4	
Size of the foreground object	40 - 70 pixels
Maximum number of foreground object contained in one image	3
Scaling rate (while ignoring aspect ratio)	80 - 120%
Gaussian noise σ	0.0 - 9.0
Gamma transform γ	0.5 - 1.5
Gaussian blur, σ	1.0 - 3.0
Gaussian blur, filter size	3 - 11
Erasing area ratio	2 - 30%

(b, For EfficientNet)	
Size of the foreground object	(B0) 149 - 187 pixels, (B1) 160 - 200 pixels, (B2) 173 - 217 pixels, (B3) 200 - 250 pixels
Maximum number of foreground object contained in one image	1
Scaling rate (while ignoring aspect ratio)	80 - 120%
Gaussian noise σ	0.0 - 9.0
Gamma transform γ	0.5 - 1.5
Gaussian blur, σ	1.0 - 3.0
Gaussian blur, filter size	3 - 11
Erasing area ratio	2 - 40%

(a) (b) (c)

Figure 3: Examples of data augmentation for the first stage using the YOLOv4. Training data were synthesized using foreground pictures of flea beetles and background field images. The squares in the panels (a) and (b) indicated a randomly erased region. The panel (c) exemplifies a synthesized image without random erasing.

a single background image. After the foreground and background images were combined, additional image processing operations were conducted using the Gaussian noise, gamma transform, and Gaussian to further diversify the synthesized images (e.g., Fig. 3c). In addition to these classical methods of data augmentation, we finally applied random erasing methods to the dataset (e.g., Fig. 3a, b). Random erasing generated training images with various levels of occlusion by randomly selecting a rectangle region in an image whose pixels were erased with random values (Zhong et al., 2020). In a captured image in a field environment, insects were partially hidden by self-occlusion, plant leaves, and soil particles. By conducting random erasing to insect-specific issues, we reduced the risk of network overfitting and made the model robust to occlusion. For the data augmentation for training YOLOv4, the parameters are shown in Table 1a.

(a) (b) (c)

Figure 4: Examples of data augmentation for the second stage using the EfficientNet. (a) A foreground image of *P. striolata* was synthesized on a background image with random erasing; (b) A foreground image of *P. atra* was synthesized on a background image; (c) A background image in which only the image processing method (Table 1b) was applied.

Second, we proposed data augmentation to train the EfficientNet as a classifier. The basic concept of this procedure was the same as that in the data extension for YOLO, where we combined a masked foreground image with a background image that did not contain any insects. At most, one foreground image was composited. For the foreground, background, and combined images, the same image processing as the data augmentation for YOLOv4 was applied. For the data augmentation employed to train the EfficientNet, the parameters are shown in Table 1b. Figure 4 shows an example of data augmentation for the second stage of the proposed augmentation method.

Insect candidate detection using a CNN-based object detector

In the emerging area of deep learning, effective and efficient CNN-based object detection models have been proposed so far, including faster R-CNN (Ren et al., 2017), Region-based Fully Convolutional Networks (R-FCN) (Dai et al., 2016), Single Shot multibox Detector (SSD) (Liu et al., 2016), and YOLO (Redmon et al., 2016). Two-stage detectors such as faster R-CNN and R-FCN had a relatively high accuracy of detection but unrealistic computational costs. By contrast, one-stage detectors such as YOLO and SSD had a lower accuracy but had an increased speed.

Redmon et al. (2016) proposed YOLO for the unified detector casting object detection as a regression problem from image pixels to spatially separated bounding boxes and associated class probabilities. Unlike region-based approaches, YOLO lacked the region proposal generation stage. Therefore, YOLO directly detected objects using a small set of candidate regions. In particular, YOLO divided an image into an $S \times S$ grid, from which C class probabilities, B bounding box locations, and confidence scores were predicted.

Bochkovskiy et al. (2020) created YOLOv4, which was an improved version of YOLO. In YOLOv4, the CNN-based feature extractor was replaced by the CSPDarknet53 feature extractor. YOLOv4 had achieved excellent scores on standard detection tasks and balanced speed and accuracy well. However, YOLOv4 might fail in detecting small objects, plausibly because of the coarse grid division and because each grid cell could only contain a few objects.

Given the disadvantage of YOLO in detecting small objects, we used YOLOv4 as a region proposal network during the first stage of the two-stage method. For each image, YOLOv4 generated a set of bounding boxes with their confidence scores ranging from 0 to 1. Confidence

score pertained to the level of confidence on whether the box contained the object. To screen insect regions, we tried the threshold of confidence score to a low value of 0.2 or 0.1. This led to the over-detection of background objects and insects. In comparison, the threshold of confidence score for an object detection task by the YOLO series was often set to a value that ranged from 0.3 to 0.5 (Padilla et al., 2021; Ovchinnikova et al., 2021; Redmon et al., 2016; Redmon and Farhadi, 2018).

Insect identification using a CNN-based classifier

In the second stage of the proposed method (Fig. 1), the candidate regions detected by YOLOv4 were re-identified as insects or not. We employed the EfficientNet as a classifier during the second stage.

The past development of CNN was synonymous to the increased depth of the network. By increasing the depth and widening the channel size, a more complex network achieved a high resolution of the image data. In addition, it resulted in more fine-grained characteristics. This development in the network size improved the classification accuracy of the network, but also led to the problem of the high computational cost of the gradient explosion parameter. ResNet (He et al., 2015) proposed that skip connection could avoid gradient explosion skillfully. MobileNet (Howard et al., 2017) used pointwise and depthwise convolutions to reduce network parameters and improve training efficiency. The SENet (Hu et al., 2017) weighted various features by loss of network training in order to achieve better results in model training.

The EfficientNet integrated characteristics of the above networks through uniformly scaling all dimensions of depth-width resolution using a compound coefficient. Unlike conventional practices that arbitrarily scaled these factors, the EfficientNet scaling method uniformly scaled the network width, depth, and resolution with a set of fixed scaling coefficients. The formula for calculating the composite proportion coefficient was as follows.

$$\begin{aligned} \text{depth} : d &= \alpha^\phi \\ \text{width} : w &= \beta^\phi \\ \text{resolution} : r &= \gamma^\phi \\ \text{s.t. } \alpha \cdot \beta^2 \cdot \gamma^2 &\approx 2 \\ \alpha \geq 1, \beta \geq 1, \gamma \geq 1 \end{aligned} \quad (1)$$

where α, β, γ were constant coefficients determined by a small grid search on the original small model. ϕ was a user-defined coefficient that controlled the number of resources that were available for model scaling. The compound scaling method was justified by the intuition that if the input image was bigger, more layers and channels were needed by the network to increase the receptive field and to capture more fine-grained patterns on the bigger image, respectively. There were eight types of the EfficientNet, from the EfficientNet-B0 to the EfficientNet-B7, with an increasing network scale.

Because the size of the candidate regions depended on the output of YOLOv4, an input image for the EfficientNet was resized to the required size of the EfficientNet irrespective of the aspect ratio. To re-identify the insects with an over-detection of the suppressed background region, the proposed EfficientNet had three corresponding output classes: *P. striolata*, *P. atra*, and the background. The Softmax function was used for the output layer of the re-identification task that posed a multi-class classification problem. A standard Softmax cross-entropy was used as a loss function.

Model evaluation

To demonstrate the effectiveness of the proposed two-stage method for the insect detection, we first applied YOLOv4 alone for detection, and then compared the results of the first stage with those of the two-stage method. For data augmentation, we evaluated the effectiveness of random erasing.

We adopted a sliding window approach to input the full captured frames into YOLOv4. We chose 512×512 pixels as an input image size for YOLO. When the image was resized to 512×512 pixels (where the frame size of the original captured image was 1,920×1,080 pixels), insects that were 2 mm in length were too compacted that even human experts could not identify them. To solve this issue, we cut an original captured image into several pieces. Considering a balance between the resolution and processing time, we split a 1920×1080-pixel image into eight 800×800 pixel pieces with overlap allowed, resulting in less than half processing cost compared to 512×512-pixel resolution of 18 split images (= 6 width × 3 height with overlap allowed). We then inputted them into the YOLOv4 (Fig. 5). Each cropped image was then resized to 512×512 pixels to fit within the required input size for YOLOv4.

To train the YOLOv4, we prepared 12,000 images using the data augmentation method proposed above. Out of all the augmented images, 80% of the images were used for training, while the remaining 20% was for validation. The YOLOv4 was trained in 150 epochs. The batch size was set at 64. Because the flea beetles were small, the training images mostly consisted of background objects. To avoid setting an infeasible number of non-target classes, we focused only on the two classes i.e., *P. striolata* or *P. atra* in the first stage. The corresponding coordinates of the bounding box were calculated after the detection for the YOLO outputs to keep the original resolution. During the sliding window approach, candidate regions having the Intersection over Union (IoU) larger than 0.3 were considered duplicates and omitted before the re-identification below.

To re-identify the candidate regions, we inputted outputs from the YOLOv4 into the EfficientNet. As mentioned above, there were eight types of the EfficientNet. Starting from the baseline EfficientNet-B0, the baseline network was scaled up from EfficientNet-B1 to B7 based on the compound scaling method with different ϕ values. The EfficientNet-B0 had the smallest network size, as it had an input image resolution of 224×224. The size of the candidate

(a)

(b)

Figure 5: Sliding window approach was used to input data to the YOLOv4. An original image (1,920×1,080 pixels) was split into eight images (800×800 pixels), and overlap was allowed. The images were then resized to 512×512 pixels for input to YOLOv4.

region detected by the YOLOv4 roughly ranged from 40×40 to 150×150 pixels; therefore, all detected candidate regions were resized to the input size of the EfficientNet regardless of the aspect ratio. When the target image was small, re-identification using a large network did not improve the accuracy but merely increased the computational cost. In this study, we chose four relatively small-sized networks (i.e., EfficientNet-B0, -B1, -B2, -B3) as a classifier. We also compared the four sizes of the EfficientNet with another classifier, Xception (Chollet, 2016). Xception (meaning "Xtreme Inception v3") was an extension of the inception architecture that replaced the standard inception modules with depth-wise separable convolutions.

To train the EfficientNet, we prepared 4,000 images for each category using the data augmentation method proposed above. We used 80% of the 4,000 images for training and the remaining 20% for validation. The EfficientNet was trained by the Adagrad in 100 epochs. The dropout rate for each task was set at 0.5. The batch size was set at 20.

All evaluations were performed on the NVIDIA GeForce RTX 2080Ti GPU with 11G memory, and the software was Windows 10, Python 3.7, CUDA 10.2. Keras 2.4.3 were used to build all models.

To compare model results with the test video, we used IoU as the threshold for positives or negatives based on the object confidence scores estimated by the object detector. IoU was defined as the intersection over the union of the two bounding boxes, which were the ground-truth bounding box and the predicted bounding box. In this paper, we set the threshold of IoU at 0.3 to consider correct or incorrect detection results.

Results

To detect candidate insect regions from the annotated test video, we first used YOLOv4 with relatively low thresholds of its confidence score. When we applied the YOLOv4 with random erasing and the threshold of the confidence score at 0.2, we obtained 489 candidate insect regions from the YOLOv4, but 44.0% of the candidate regions was false positive (Table S1a) and thus considered over-detection. The over-detection was observed even when random erasing was not applied (Table S1b). These results were plausible because the threshold of confidence score was relatively low for detecting as many small insects as possible. To obtain more candidate regions, we further decreased the threshold of the confidence score to 0.1. The lower threshold led us to detect additional 259 regions of background objects but detected only two additional regions of *P. atra* (Table S1c), indicating little improvement of insect detection with

an increased computational cost for background objects. Therefore, YOLOv4 was used with the threshold of the confidence score at 0.2 in the following analyses. In contrast, if over-detection was unfavorable, it could be suppressed by setting a threshold higher than 0.2. However, the undetected rate would be even worse than the current undetected rate of 58.0% (Table S1a).

Using the annotated test video (Supplementary video S1), we then compared the performance of the YOLOv4 with or without random erasing. Table 2 displays the confusion matrices of the one-stage detection using only the YOLOv4 with/without random erasing, where the rows and columns of the confusion matrix correspond to the true classes and predicted classes, respectively. To quantify the detection performance, we calculated the precision, recall, and F-measure (Table 3). The results showed that the detection performance on the test video was moderately improved by subjecting the images to random erasing (Table 3). Both the trained YOLOv4 models with and without random erasing were well fitted to validation datasets (mean Average Precision approached nearly 100%; Figure S1), indicating that the YOLOv4 model was well trained irrespective of random erasing.

To re-identify candidate regions, we then applied EfficientNet for the outputs from the YOLOv4. Table 4 shows the results in which outputs from the YOLOv4 with random erasing were re-identified using the EfficientNet. In these results, all classifier models were trained on the dataset with random elimination applied. The results suggest that EfficientNet-B1 was the most accurate among the five classifiers (Table 4). The trained EfficientNet-B1 model was well fitted to validation datasets (Figure S2), indicating that the model training operated well.

To test the effects of random erasing on the performance of the EfficientNet, we applied the two-stage method for the annotated test video. Table 5 displays the confusion matrices of the two-stage detection using the YOLOv4 with random erasing and the EfficientNet-B1 with or without random erasing. Table 6 shows the detection performance of these methods. By applying random erasing to the training dataset of the EfficientNet, performance was also moderately improved (Table 6).

After re-identification by the EfficientNet-B1, 87.9% of falsely detected background areas were corrected as background classes (Table 2 and 6; Table S1 and S2; Supplementary Video S2). Thus, the two-stage detection greatly improved the precision compared with one-stage detection using only the YOLOv4. We also confirmed that the precision and resultant F-value were improved, while the recall was nearly the same (Table 3 and 6).

Finally, we evaluated the speed to detect and re-identify a candidate region. Table 7 shows the processing time of re-identification using each classifier model for one candidate region. In this table, the detection with YOLOv4 and re-identification corresponded to the frame rate, which was the frame per second (FPS) of the detection process for one frame and the re-identification process for two candidate

Table 2

Confusion matrices of the one-stage detection using only the YOLOv4 with (a) or without (b) random erasing. The rows and columns correspond to the ground truth and prediction, respectively.

(a) With random erasing		
	<i>P. striolata</i>	<i>P. atra</i>
<i>P. striolata</i>	171	4
<i>P. atra</i>	0	99
Background	93	122

(b) Without random erasing		
	<i>P. striolata</i>	<i>P. atra</i>
<i>P. striolata</i>	159	13
<i>P. atra</i>	4	103
Background	115	137

Table 3

Detection results using only the YOLOv4 with/without Random Erasing (R. E.).

	Precision	Recall	F-measure
YOLOv4 w/ R. E.	0.55	0.42	0.48
YOLOv4 w/o R. E.	0.49	0.41	0.45

Table 4

Comparison of performance among the EfficientNet and Xception models.

	Precision	Recall	F-measure
EfficientNet-B0	0.86	0.38	0.53
EfficientNet-B1	0.89	0.41	0.56
EfficientNet-B2	0.89	0.40	0.55
EfficientNet-B3	0.91	0.39	0.55
Xception	0.84	0.35	0.49

Figure 7: Examples of the smallest individuals that were detected correctly. The size of the original image is 1,920×1,080 pixels, where the regions of flea beetles detected are ca. 35×50 pixels. The video results are available as a supplementary material (Supplementary Video S3).

regions since the number of candidate regions detected from a single frame in this evaluation was around two.

(a) (b) (c)

Figure 8: Examples of misidentified *P. atra* images by EfficientNet-B1. (a) Blurred *P. atra* image misidentified as a background object; (b) Clipped *P. atra* image misidentified as a background object; (c) Light-reflected *P. atra* image misidentified as *P. striolata*

(a) Before

(b) After

Figure 6: Example of correct identification by re-identification. Results of the first stage using the YOLOv4 (a), and results of the second stage using the EfficientNet (b). The video results of the two-stage method are available as a supplementary material (Supplementary Video S2).

Table 5

Confusion matrices of the two-stage detection using the EfficientNet-B1 with (a) or without (b) random erasing. The rows and columns correspond to the ground truth and prediction, respectively.

(a) With random erasing			
	<i>P. striolata</i>	<i>P. atra</i>	Background
<i>P. striolata</i>	170	3	2
<i>P. atra</i>	2	90	7
Background	15	11	189

(b) Without random erasing			
	<i>P. striolata</i>	<i>P. atra</i>	Background
<i>P. striolata</i>	164	4	7
<i>P. atra</i>	5	84	10
Background	20	15	180

Table 6

Detection results the two-stage detection with/without Random Erasing (R. E.).

	Precision	Recall	F-measure
EfficientNet-B1 w/ R. E.	0.89	0.41	0.55
EfficientNet-B1 w/o R. E.	0.85	0.39	0.53

Table 7

Comparison of the processing time among the re-identification models in seconds (s) and frames per second (FPS).

Re-identification models	Classification of one image [s]	Detection with YOLOv4 and re-identification [FPS]
EfficientNet-B0	0.027	5.4
EfficientNet-B1	0.029	5.3
EfficientNet-B2	0.030	5.3
EfficientNet-B3	0.033	5.1
Xception	0.031	5.2

Discussion

The paradigm of visual recognition has changed rapidly after the emergence of the ImageNet dataset, which has demonstrated the power of data-driven feature learning. During the past years, CNN has become the most effective architecture for visual recognition tasks. Therefore, many researches have been conducted on deep learning-based approaches for insect detection and identification (Lim et al., 2017; Rustia et al., 2018; Shen et al., 2018; Xia et al., 2018;

He et al., 2019). Recent studies have proposed hierarchical CNNs that consist of a detector and classifier that can distinguish small arthropods and soil particles on a supplementary sheet under controlled conditions (Tresson et al., 2019, 2021). To achieve a system that is more suitable for field use and more user-friendly, we advocated the usage of a unique application of CNN-based object detection to a video collected by a handy camera in the field. The present study provided a two-stage method that could greatly improve the precision to distinguish small congeners and background objects, while there remained major challenges in real field environments regarding the size, resemblance, and training data of insects.

Given the advantage of YOLO in rapid processing, we aimed to create a balance between the speed and accuracy in our two-stage method. The processing of the proposed two-stage method is approximately 5 FPS in total (Table 7), which suggests an acceptable speed in an interactive user interface. As if unfamiliar languages are translated using a smartphone camera, we suppose that users leverage the proposed system to quickly check the location and species of insects in the field. Furthermore, sedentary insects do not suddenly move out of the view of a camera. Unless the target is a fast-moving species such as an adult butterfly flying across the sky, the processing cost of the proposed method is unlikely to be a hindrance for practical use. While the current performance (namely 5 FPS) is not so smooth that human recognizes the output as a real-time video, it would be faster due to technological advances in the future.

One of the most serious issues is the difficulty in distinguishing small insects and background objects. Even though the YOLOv4 alone could distinguish two species of flea beetles, we observed over-detection simultaneously (Table 2); thus, it was plausible to use an additional CNN-based classifier to suppress over-detection. Combined with the latest object detection methods, the proposed two-stage method correctly re-identified most of the over-detections as background (Table 5). Figure 6 shows an example of correct identification by re-identification, where background objects (Fig. 6a upper) or a leaf hole (Fig. 6a lower) are initially mistaken as flea beetles but are then re-identified as background. Notably, the smallest individual that could be detected by the proposed method was 35×50 pixels in the HD size image (Fig. 7). The area ratio of the detected insect to frames was merely <0.1%. These results suggest that our two-stage method has a potential to distinguish small targets from various background objects.

Another issue in the field is heterogeneous photographing conditions due to lighting and angles. While the precision was greatly improved, the recall slightly decreased due to the re-identification. This is because the second stage cannot detect more objects than the first stage, and it cannot classify candidate regions perfectly. Specifically, Figure 8 exemplifies misidentified images, including blurred, partially clipped, or reflected *P. atra*. Although mechanisms through which CNN-based methods classify each image are generally unknown, the misidentified images highlight small inconspicuous insects under the field condition as hard-to-identify objects. Furthermore, Figure 6 shows that a *P. atra* individual located in the lower center cannot be detected until it moves (until the 30 s mark in Supplementary Video S3). This missed detection might have occurred due to the strong light source, which was the sunlight, was reflected on *P. atra*'s black back with a small contrast between the target i.e., the black beetle and the background, which consisted of dark-colored leaves. Figure 7 also included an overlooked *P. atra* as it was small and blurred on the edge of a black pot, which could not be detected at all after the 36 s mark in Supplementary Video S3. This series of undetected *P. atra* individuals likely resulted in the small recall in the present task (Table S2).

Although the small recall is unsuitable for automatic counting using a fixed-point camera, the performance of our method is still valuable when implementing an interactive user interface for non-experts. In fact, even the undetected *P. atra* individual was finally detected when its appearance changed among different frames (from the 31 s mark in Supplementary Video S3). Since insects moved spontaneously, they exhibited various appearances in different times. It was also possible that movements of a camera device, including camera shaking and resultant re-focusing, led to slight changes in insect appearance. While changes in the camera position might have resulted in blurring and other errors, they also provided an opportunity to capture another appearance of the target object. We suppose that the present system can be used by non-experts, who are not nature photographers and entomologists. Therefore, our system possesses a merit for non-experts to easily change the viewpoint of a handy camera and thereby screen an individual insect from one of the multiple frames. Far lower thresholds of confidence score would work better with YOLO for the other tasks, though the 0.2 threshold was considered optimal in the present task (Table S1).

The latest high-performance CNN-based detection and classification models require a large number of images to optimize their numerous parameters. Only a few practical and large datasets currently exist for insect detection because setting many cameras to capture numerous insects in various field backgrounds is costly and laborious (Preti et al., 2020). Although we collected hundreds of flea beetle images, merely thirty images were considered effective samples for each species. A large dataset was generated by applying data augmentation based on standard image processing such as inversion, rotation, blurring, and chroma

key synthesis. Furthermore, we applied random erasing to anticipate an occlusion of insects in the real environment. Although camera devices were different between the training images and test video, the random erasing contributed to the improvement of the accuracy in both the detector and classifier. The proposed image enhancement based on chroma key synthesis and random erasing would help us deal with major obstacles of data collection in research using CNNs for insect detection and identification.

Our two-stage method for tiny flea beetles may be widely applied to various insect herbivores and plants. Diverse insect herbivores, such as beetles, caterpillars, aphids and thrips, occur on *Brassica* cultivars and wild *Arabidopsis* species (Ahuja et al., 2010; Sato et al., 2019; Matsubara and Sugiura, 2021; Shimizu-Inatsugi et al., 2021). Many of these crucifer-feeding insects are as small as or larger than *Phyllotreta* beetles. For example, lepidopterous herbivores (e.g., *Plutella xylostella* and *Pieris rapae*) are recognized as a serious pest of *Brassica* crops (Ahuja et al., 2010). Last-instar larvae of them are larger than 0.5 cm and more sedentary than *Phyllotreta* beetles. In addition, *Brassica* crops are larger than our study species, *A. thaliana*. The two-stage method is therefore expected to be more feasible for large herbivores and plants than the *Phyllotreta-Arabidopsis* system. In contrast, thrips and aphids are smaller than *Phyllotreta* beetles and are likely to create a clonal colony including multiple individuals. Overlap of individuals is another issue when applying our method to the colony of small insects.

Conclusion

To overcome the difficulty in detecting tiny insects in the field, we proposed a two-stage detection method with data augmentation. This approach improved the precision of an existing CNN-based method to help detect and identify insects in the field. To screen small insects, the YOLOv4 was used as a detector. Once candidate regions were specified, another CNN-based object classification model, the EfficientNet, successfully re-identified the congeneric species from the candidates. Now that CNN-based methods have garnered much attention in biodiversity research (Norouzzadeh et al., 2018; Willi et al., 2019; Høye et al., 2021), we could combine such a state-of-the-art method to facilitate automatic identification of diverse organisms in the field.

Declaration of Competing Interest

None.

Acknowledgements

The authors would like to express their sincerest gratitude to H. Kuzuhara and T. Kano for the development of analytical pipelines; K. Takeda for the insect annotation; and to all members of the Shimizu group for helping with the set-up of plants in the field. This study was supported by the Japan Science and Technology Agency (JST) through the PRESTO

(No. JPMJPR17Q4 to YS) and CREST (No. JPMJCR15O2 and JPMJCR16O3 to AJN and KKS) projects, the Japan Society for the Promotion Science (JSPS) KAKENHI (No. 20K15880 to YS), the Swiss National Science Foundation (No. 31003A_182318 to KKS), and the University Research Priority Project (URPP) Global Change and Biodiversity of the University of Zurich to KKS.

References

Ahuja, I., Rohloff, J., Bones, A.M., 2010. Defence mechanisms of brassicaceae: implications for plant-insect interactions and potential for integrated pest management. a review. *Agronomy for sustainable development* 30, 311–348.

Almryad, A.S., Kutucu, H., 2020. Automatic identification for field butterflies by convolutional neural networks. *Engineering Science and Technology, an International Journal* 23, 189–195.

Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M., 2020. YOLOv4: Optimal Speed and Accuracy of Object Detection. *arXiv e-prints*, arXiv:2004.10934arXiv:2004.10934.

Chollet, F., 2016. Xception: Deep learning with depthwise separable convolutions. *Cite arxiv:1610.02357*.

Dai, J., Li, Y., He, K., Sun, J., 2016. R-fcn: Object detection via region-based fully convolutional networks, in: *Proceedings of the 30th International Conference on Neural Information Processing Systems*, Curran Associates Inc., Red Hook, NY, USA. pp. 379–387.

Deng, L., Wang, Y., Han, Z., Yu, R., 2018. Research on insect pest image detection and recognition based on bio-inspired methods. *Biosystems Engineering* 169, 139–148. doi:10.1016/j.biosystemseng.2018.02.008.

Hansen, O.L., Svenning, J.C., Olsen, K., Dupont, S., Garner, B.H., Iosifidis, A., Price, B.W., Høye, T.T., 2020. Species-level image classification with convolutional neural network enables insect identification from habitus images. *Ecology and Evolution* 10, 737–747.

He, K., Zhang, X., Ren, S., Sun, J., 2015. Deep residual learning for image recognition. *CoRR abs/1512.03385*. URL: <http://arxiv.org/abs/1512.03385>, arXiv:1512.03385.

He, Y., Zeng, H., Fan, Y., Ji, S., Wu, J., 2019. Application of deep learning in integrated pest management: A real-time system for detection and diagnosis of oilseed rape pests. *Mobile Information Systems* 2019, 4570808:1–4570808:14.

Hogeweg, L., Zeegers, T., Katramados, I., Jongejans, E., 2019. Smart insect cameras. *Biodiversity Information Science and Standards* 3, e39241.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H., 2017. Mobilenet: Efficient convolutional neural networks for mobile vision applications. URL: <http://arxiv.org/abs/1704.04861>. cite arxiv:1704.04861.

Høye, T.T., Årje, J., Bjerger, K., Hansen, C., Iosifidis, A., Leese, F., Mann, H.M., Meissner, K., Melvad, C., Raitoharju, J., 2021. Deep learning and computer vision will transform entomology. *Proceedings of the National Academy of Sciences USA* 118.

Hu, J., Shen, L., Albanie, S., Sun, G., Wu, E., 2017. Squeeze-and-excitation networks. URL: <http://arxiv.org/abs/1709.01507>. cite arxiv:1709.01507Comment: journal version of the CVPR 2018 paper, accepted by TPAMI.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. *Nature* 521, 436–444. doi:10.1038/nature14539.

Lim, S., Kim, S., Kim, D., 2017. Performance effect analysis for insect classification using convolutional neural network. 2017 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE), 210–215.

Liu, L., Ouyang, W., Wang, X., Fieguth, P., Chen, J., Liu, X., Pietikäinen, M., 2019. Deep learning for generic object detection: A survey. *International Journal of Computer Vision* 128, 261–318.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S.E., Fu, C.Y., Berg, A.C., 2016. SSD: Single shot multibox detector, in: *Leibe, B., Matas, J., Sebe, N., Welling, M. (Eds.), ECCV (1), Springer*. pp. 21–37. doi:10.1007/978-3-319-46448-0_2.

Martineau, M., Conte, D., Raveaux, R., Arnault, I., Munier, D., Venturini, G., 2017. A survey on image-based insect classification. *Pattern Recognition* 65, 273–284.

Matsubara, S., Sugiura, S., 2021. A Technique for Multi-Generational Rearing of *Phaedon brassicae* (Coleoptera: Chrysomelidae). *Entomological News* 129, 431–434. doi:10.3157/021.129.0413.

Mayo, M., Watson, A.T., 2007. Automatic species identification of live moths. *Knowledge-Based Systems* 20, 195–202. doi:10.1016/j.knosys.2006.11.012. al 2006.

Norouzzadeh, M.S., Nguyen, A., Kosmala, M., Swanson, A., Palmer, M.S., Packer, C., Clune, J., 2018. Automatically identifying, counting, and describing wild animals in camera-trap images with deep learning. *Proceedings of the National Academy of Sciences USA* 115, E5716–E5725.

Ovchinnikova, K., James, M.A., Mendo, T., Dawkins, M., Crall, J., Boswarva, K., 2021. Exploring the potential to use low cost imaging and an open source convolutional neural network detector to support stock assessment of the king scallop (*Pecten maximus*). *Ecological Informatics* 62. doi:10.1016/j.ecoinf.2021.101233.

Padilla, R., Passos, W., Dias, T.L.B., Netto, S.L., da Silva, E.A.B., 2021. A comparative analysis of object detection metrics with a companion open-source tool. *Electronics* 10. URL: <https://www.mdpi.com/2079-9292/10/3/27>. doi:10.3390/electronics10030279.

Park, J., Kim, D., Choi, B., Kang, W., Kwon, H.W., 2020. Classification and morphological analysis of vector mosquitoes using deep convolutional neural networks. *Scientific reports* 10, 1–12.

Preti, N., Verjegen, F., Angeli, S., 2020. Insect pest monitoring with camera-equipped traps: strengths and limitations. *Journal of Pest Science*, 1–15.

Rani, R.U., Amsini, P., 2016. Pest identification in leaf images using deep neural classifier. *International Journal of Computational Intelligence and Informatics* 6, 248–260.

Kedmon, J., Divvala, S., Girshick, R., Farhadi, A., 2016. You only look once: Unified, real-time object detection, in: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 779–788. doi:10.1109/CVPR.2016.91.

Redmon, J., Farhadi, A., 2018. YoloV3: An incremental improvement. *ArXiv abs/1804.02767*.

Ren, S., He, K., Girshick, R., Sun, J., 2017. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 1137–1149. doi:10.1109/TPAMI.2016.2577031.

Rustia, D.J.A., Lin, C.E., yung Chung, J., 2018. A real-time multi-class insect pest identification method using cascaded convolutional neural networks, in: *Proceedings of 9th International Symposium on Machinery and Mechatronics for Agriculture and Biosystems Engineering*, p. 67.

Samanta, R.K., Ghosh, I., 2012. Tea insect pests classification based on artificial neural networks. *International Journal of Computer Engineering Science* 2, 1–13.

Sato, Y., Shimizu-Inatsugi, R., Yamazaki, M., Shimizu, K.K., Nagano, A.J., 2019. Plant trichomes and a single gene *GLABRA1* contribute to insect community composition on field-grown *Arabidopsis thaliana*. *BMC Plant Biology* 19. doi:10.1186/s12870-019-1705-2.

Shen, Y., Zhou, H., Li, J., Jian, F., Jayas, D.S., 2018. Detection of stored-grain insects using deep learning. *Computers and Electronics in Agriculture* 145, 319–325. doi:10.1016/j.compag.2017.11.039.

Shimizu-Inatsugi, R., Milosavljevic, S., Shimizu, K.K., Schaepman-Strub, G., Tanoi, K., Sato, Y., 2021. Metal accumulation and its effect on leaf herbivory in an allopolyploid species *Arabidopsis kamchatica* inherited from a diploid hyperaccumulator *A. halleri*. *Plant Species Biology* 36, 208–217. doi:10.1111/1442-1984.12304.

Shorten, C., Khoshgoftaar, T., 2019. A survey on image data augmentation for deep learning. *Journal of Big Data* 6, 1–48.

Silvertown, J., Harvey, M., Greenwood, R., Dodd, M., Rosewell, J., Rebelo, T., Ansine, J., McConway, K., 2015. Crowdsourcing the identification of organisms: A case-study of iSpot. *ZooKeys*, 125.

Tan, M., Le, Q., 2019. EfficientNet: Rethinking model scaling for convolutional neural networks, in: *Chaudhuri, K., Salakhutdinov, R. (Eds.), Proceedings of the 36th International Conference on Machine Learning*,

PMLR. pp. 6105–6114.

- Tresson, P., Carval, D., Tixier, P., Puech, W., 2021. Hierarchical classification of very small objects: Application to the detection of arthropod species. *IEEE Access* 9, 63925–63932.
- Tresson, P., Tixier, P., Puech, W., Bagny Beilhe, L., Roudine, S., Pages, C., Carval, D., 2019. Corigan: Assessing multiple species and interactions within images. *Methods in Ecology and Evolution* 10, 1888–1893.
- Valan, M., Makonyi, K., Maki, A., Vondráček, D., Ronquist, F., 2019. Automated taxonomic identification of insects with expert-level accuracy using effective feature transfer from convolutional networks. *Systematic Biology* 68, 876–895.
- Van Etten, A., 2018. You only look twice: Rapid multi-scale object detection in satellite imagery. *arXiv preprint arXiv:1805.09512*.
- Van Horn, G., Mac Aodha, O., Song, Y., Cui, Y., Sun, C., Shepard, A., Adam, H., Perona, P., Belongie, S., 2018. The iNaturalist species classification and detection dataset, in: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 8769–8778.
- Venugoban, K., Ramanan, A., 2014. Image classification of paddy field insect pests using gradient-based features. *International Journal of Machine Learning and Computing*, 1–5.
- Wang, J., Lin, C., Ji, L., Liang, A., 2012. A new automatic identification system of insect images at the order level. *Knowledge-Based Systems* 33, 102–110. doi:10.1016/j.knsys.2012.03.014.
- Willi, M., Pitman, R.T., Cardoso, A.W., Locke, C., Swanson, A., Boyer, A., Veldhuis, M., Fortson, L., 2019. Identifying animal species in camera trap images using deep learning and citizen science. *Methods in Ecology and Evolution* 10, 80–91.
- Xia, D., Chen, P., Wang, B., Zhang, J., Xie, C., 2018. Insect detection and classification based on an improved convolutional neural network. *Sensors* 18. doi:10.3390/s18124169.
- Xie, C., Wang, R., Zhang, J., Chen, P., Dong, W., Li, R., Chen, T., Chen, H., 2018. Multi-level learning features for automatic classification of crop pests. *Computers and Electronics in Agriculture* 152, 23–241. doi:10.1016/j.compag.2018.07.014.
- Xie, C., Zhang, J., Li, R., Li, J., Hong, P., Xia, J., Chen, P., 2015. Automatic classification for field crop insects via multiple-task sparse representation and multiple-kernel learning. *Computers and Electronics in Agriculture* 119, 123–132. doi:10.1016/j.compag.2015.10.015.
- Yalcin, H., 2015. Vision based automatic inspection of insects in pheromone traps, in: *2015 Fourth International Conference on Agro-Geoinformatics (Agro-geoinformatics)*, pp. 333–338. doi:10.1109/Agro-Geoinformatics.2015.7248113.
- Yamashita, R., Nishio, M., Do, R., Togashi, K., 2018. Convolutional neural networks: an overview and application to radiology. *Insights into Imaging* 9, 611–629. doi:10.1007/s13244-018-0639-9.
- Zhong, Z., Zheng, L., Kang, G., Li, S., Wang, Y., 2020. Random erasing data augmentation, in: *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, AAAI Press*. pp. 13001–13008. doi:10.1609/aaai.v34i07.7000.

Supplementary Materials

Supplementary Video S1-S3 and photographs generated by our own fieldworks are available at the GitHub repository (<https://github.com/h-taki/identification-of-tiny-insects>)

Highlights

Using a two-stage convolutional neural network to rapidly identify tiny herbivorous beetles in the field

Hironori Takimoto, Yasuhiro Sato, Atsushi J. Nagano, Kentaro K. Shimizu, Akihiro Kanagawa

- We proposed a two-stage convolutional neural network (CNN) method in this study.
- In this method, YOLOv4 and EfficientNet were used for the detection of small insects.
- The two-stage method was applied to a field video of 2-mm long beetles.
- Our method classified the beetles and backgrounds more precisely than did YOLO alone.

Fig. 1 Flow of the proposed two-stage detection method. During the first stage (left), the proposed method detected several candidate regions from a captured image by YOLOv4 with a low threshold of detection. In the second stage (right), all candidate regions were re-identified using the EfficientNet.

Fig. 2 Example of a captured image in a real field environment. Two species of *Phyllotreta* beetles (red square; ca. 2 mm in body length) simultaneously appeared on an individual *Arabidopsis thaliana*.

Fig. 3 Examples of data augmentation for the first stage using the YOLOv4. Training data were synthesized using foreground pictures of flea beetles and background field images. The squares in the panels (a) and (b) indicated a randomly erased region. The panel (c) exemplifies a synthesized image without random erasing.

Fig. 4 Examples of data augmentation for the second stage using the EfficientNet. (a) A foreground image of *P. striolata* was synthesized on a background image with random erasing; (b) A foreground image of *P. atra* was synthesized on a background image; (c) A background image in which only the image processing method ([Table 1b](#)) was applied.

Fig. 5 Sliding window approach was used to input data to the YOLOv4. An original image (1920×1080 pixels) was split into eight images (800×800 pixels), and overlap was allowed. The images were then resized to 512×512 pixels for input to YOLOv4.

Fig. 6 Example of correct identification by re-identification. Results of the first stage using the YOLOv4 (a), and results of the second stage using the EfficientNet (b). The video results of the two-stage method are available as a supplementary material (Supplementary Video S2). (a) Before. (b) After.

Fig. 7 Examples of the smallest individuals that were detected correctly. The size of the original image is 1920×1080 pixels, where the region of flea beetles detected are ca. 35×50 pixels. The video results are available as a supplementary material (Supplementary Video S3).

Fig. 8 Examples of misidentified *P. atra* images by EfficientNet-B1. (a) Blurred *P. atra* image misidentified as a background object; (b) Clipped *P. atra* image misidentified as a background object; (c) Light-reflected *P. atra* image misidentified as *P. striolatus*.