Computers and Electronics in Agriculture 166 (2019) 105002

Contents lists available at ScienceDirect

Computers and Electronics in Agriculture

journal homepage: www.elsevier.com/locate/compag

A deep learning model to recognize food contaminating beetle species based on elytra fragments

Leihong Wu[a], Zhichao Liu[a], Tanmay Bera[a], Hongjian Ding[b], Darryl A. Langley[b], Amy Jenkins-Barnes[b], Cesare Furlanello[c], Valerio Maggio[c], Weida Tong[a], Joshua Xu[a,*]

[a] Division of Bioinformatics and Biostatistics, National Center for Toxicological Research, U.S. Food and Drug Administration, Jefferson, AR 72079, USA
[b] Food Chemistry Laboratory-1, Arkansas Laboratory, Office of Regulatory Affairs, U.S. Food and Drug Administration, Jefferson, AR 72079, USA
[c] Predictive Models for Biomedicine and Environment Unit, Fondazione Bruno Kessler, Trento, Italy

ARTICLE INFO

Keywords:
Food contamination
Beetle species identification
Deep learning
Convolutional neural networks
Transfer learning

ABSTRACT

Insect pests are often associated with food contamination and public health risks. Accurate and timely species-specific identification of pests is a key step to scale impacts, trace back the contamination process and promptly set intervention measures, which usually have serious economic impact. The current procedure involves visual inspection by human analysts of pest fragments recovered from food samples, a time-consuming and error-prone process. Deep Learning models have been widely applied for image recognition, outperforming other machine learning algorithms; however only few studies have applied deep learning for food contamination detection. In this paper, we describe our solution for automatic identification of 15 storage product beetle species frequently detected in food inspection. Our approach is based on a convolutional neural network trained on a dataset of 6900 microscopic images of elytra fragments, obtaining an overall accuracy of 83.8% in cross validation. Notably, the classification performance is obtained without the need of designing and selecting domain specific image features, thus demonstrating the promising prospects of Deep Learning models in detecting food contamination.

1. Introduction

Food grains such as rice, wheat, corn are often contaminated by insect pests such as pantry beetles. Food products that use these gains or processed in unhygienic conditions are also prone to similar contamination (Gorham, 1979; Rees, 2004). With some species being active carriers of pathogens, these winged creatures can spread fast and quickly make large quantities of food unsafe for human consumption (James F. Champbell, 2004; Zurek and Gorham, 2008). Their spread can further be escalated if contaminated food grains and/or products are not regulated during the global transportation of food products. Such incidents can also introduce invasive foreign pests that can cause severe environmental damage. Thus, regulatory agencies, including the US-FDA and EFSA, throughout the world continuously survey food samples to monitor for this contamination in order to prevent pests for being more destructive (Stejskal et al., 2015). This requires collection and processing of hundreds of food samples and their subsequent screening for filth elements such as beetle remains, which would indicate a contamination. Unfortunately, just detecting the contamination is not adequate. Their correct identification is also required to better manage contaminations, as each species poses different levels of threat.

To date, the species identification procedure consists of visual analysis by human expert at the comparison microscope, typically of fragments of the collected specimens. However, the process is highly time-consuming, error prone and difficult to automate, especially for beetle species with similar appearance. More refined procedures combine the visual inspection of fragments with other technologies, like genomics (Wu et al., 2017). However, a challenge to food filth analysis lies in the difficulty of obtaining pure beetle DNA. Often, no body parts are recovered besides elytra, or hardened forewings, which are mainly made of proteins. Additional experiments may further increase the workload of food filth analysts and decrease efficiency. Notably, extra omics analysis may also introduce a burden to the regulatory action, increasing the overall cost of food processing.

Computer-assisted image analyses have been applied in beetles and food safety research for decades (Boyer et al., 2011; Daly et al., 1982; Gaston and O'Neill, 2004; Larios et al., 2008; Weeks et al., 1997). The approach has evolved rapidly due to the emergence of Machine Learning (ML) algorithms (Karunakaran et al., 2004; Zou et al., 2008), including Support Vector Machine (SVM) (Cortes and Vapnik, 1995),

* Corresponding author.
E-mail address: zhihua.xu@fda.hhs.gov (J. Xu).

https://doi.org/10.1016/j.compag.2019.105002
Received 29 August 2018; Received in revised form 15 July 2019; Accepted 8 September 2019

Table 1
Beetle Elytra Images Dataset Statistics used in this study.

Index Number	Genus	Species	Common Name	# whole elytra images
S01	*Cryptolestes*	*pusillus*	Flat Grain Beetle	5
S02	*Lasioderma*	*serricorne*	Cigarette Beetle	5
S03	*Gnathocerus*	*cornutus*	Broad-horned flour beetle	4
S04	*Zabrotes*	*subfasciatus*	Mexican Bean Weevil	6
S05	*Oryzaephilus*	*mercator*	Merchant Grain Beetle	4
S06	*Oryzaephilus*	*surinamensis*	Saw-toothed Grain Beetle	4
S07	*Rhyzopertha*	*dominica*	Lesser Grain Borer	4
S08	*Sitophilus*	*granarius*	Granary Weevil	5
S09	*Sitophilus*	*oryzae*	Rice Weevil	4
S10	*Stegobium*	*paniceum*	Drugstore Beetle	5
S11	*Tribolium*	*brevicornis*	North American Flour Beetle	5
S12	*Tribolium*	*castaneum*	Red Flour Beetle	3
S13	*Tribolium*	*confusum*	Confused Flour Beetle	5
S14	*Tribolium*	*freemani*	Kashmir Flour Beetle	5
S15	*Tribolium*	*madens*	Black Flour Beetle	5

and Random Forests (RF) (Breiman, 2001). These algorithms have been adopted for food contamination analysis, achieving competitive results. An artificial Neural Network (ANN) based model (Park et al., 2016) has been used for the beetle species recognition problem, achieving an average classification accuracy (ACC) of 0.8. Later, we further improved this study by adopting an SVM model (Bisgin et al., 2018), and improving the ACC to 0.85. Notably, both methods rely on a feature extraction step, namely an appropriate, and possibly discriminant set of features is derived from raw images then used as the input to predictive models, ANN and SVM, respectively. However, despite the good performance results, this feature engineering process imposes a strong limitation to the adoption of the whole pipeline to different experimental settings (e.g. non-elytra images).

Deep Learning (DL) (Goodfellow et al., 2016; LeCun et al., 2015; Schmidhuber, 2015) has drastically emerged as the paradigm of image processing, improving the state-of-the-art in many domains, including object detection and object recognition. Differently from conventional image recognition methods, deep learning architectures like Convolutional Neural Networks (CNNs) can use the raw image as input as they incorporate the feature construction step directly into the learning process, by updating their parameters and connections as a function of the error on a set of training data (LeCun et al., 2015). Deep Learning has already been applied in food and agricultural science such as identifying Northern leaf blight in Maize plants (DeChant et al., 2017), rice disease detection (Lu et al., 2017), food image classification and recognition (Kagaya and Aizawa, 2015; Kagaya et al., 2014), symptoms detection (Cardinale et al., 2018; Cruz et al., 2019, 2017), moth detection (Wen et al., 2015) and plant identification (Lee et al., 2015). Different from many other object recognition studies that focus on the entire object, identification of filth elements such as beetle remains needs to recognize the species not from the entire beetle image, but only from small fragments of the beetle elytra. However, a very few deep learning models have currently been implemented for filth elements identification in food (Bansal et al., 2017; Reinholds et al., 2015).

In this endeavor, we present an end-to-end DL based approach for beetle species identification on filth elements (elytra fragments) from food contaminations. Our solution exploits a VGG16-based model (Simonyan and Zisserman, 2014), that is one widely used neural architectures originally designed for image classification tasks, and trained on the ImageNet dataset (Deng et al., 2009). Our model is first initialized using the publicly available ImageNet-training weights. Then, all the convolutional blocks are kept unchanged, while only the top layers of the network, i.e. fully-connected layers, are trained over images of beetles. This process is generally referred to as *transfer learning* or *domain adaptation (*Bengio, 2012*)*. Our solution achieves an average ACC of 0.838 (CI: 0.829, 0.846) in the classification of the 15 beetle species frequently detected in food inspection, improving the current state of the art results. Our method therefore explores the possibility of using DL models in food filth and food safety analysis and avoids a lengthy feature extraction work, which sensibly eases the model development process. We believe our work will lay foundation to similar studies in this area in the near future.

2. Materials and methods

2.1. Beetle elytra images dataset

The Beetle Elytra Images (BEI) dataset used in this study was collected from Arkansas Laboratory. We harvested the whole elytra from a whole specimen and collected high-resolution images with no food background. Each image has been manually annotated with the corresponding species by human experts. In details, the dataset consists of 69 whole high-resolution elytra images for 15 beetle species, namely *Cryptolestes pusillus, Lasioderma serricorne, Gnathocerus cornutus, Zabrotes subfasciatus, Oryzaephilus mercator, Oryzaephilus surinamensis, Rhyzopertha dominica, Sitophilus granarius, Sitophilus oryzae, Stegobium paniceum, Tribolium brevicornis, Tribolium castaneum, Tribolium confusum, Tribolium freemani, and Tribolium madens.* Each beetle species consists of 3 to 6 high-resolution images respectively. The size and shape of the elytra vary from less than 1 mm up to 2 mm, depending the size of insect. Resulting elytra images have a width ranging from 3000 to 4000 pixels, and a height ranging from 1000 to 2000 pixels. The BEI dataset has been also used in other research studies: (Bisgin et al., 2018; Park et al., 2016). A complete summary of the dataset is provided in Table 1.

It is worth noting that some of these species share the same genus classification - e.g. *Tribolium madens*, and *Tribolium castaneum* – possibly leading to confusion when visually inspected by human analysts. However, while *T. madens* (black flour beetle) feeds on grain that is damaged or is going out of condition and it can be found in flour mills and warehouses, *T. castaneum* attacks grains that are intact and stable, but have become heated. Thus, being able to correctly distinguish the two species gives crucial indication about the infestation, even if the damage is not distinctive. Two full-body example images for the two species are shown in Fig. 1a. An example of fragmented images used in this study is shown in Fig. 1b-e. As shown, with the lab light condition, S12 and S15 showed different colors of elytra which is an important factor to differentiate them. However, there are also beetles that from different species, however had quite similar elytra fragments (S13 and S14, Fig. 1f).

2.2. Image pre-processing

To mimic the elytra remains commonly observed in contaminated

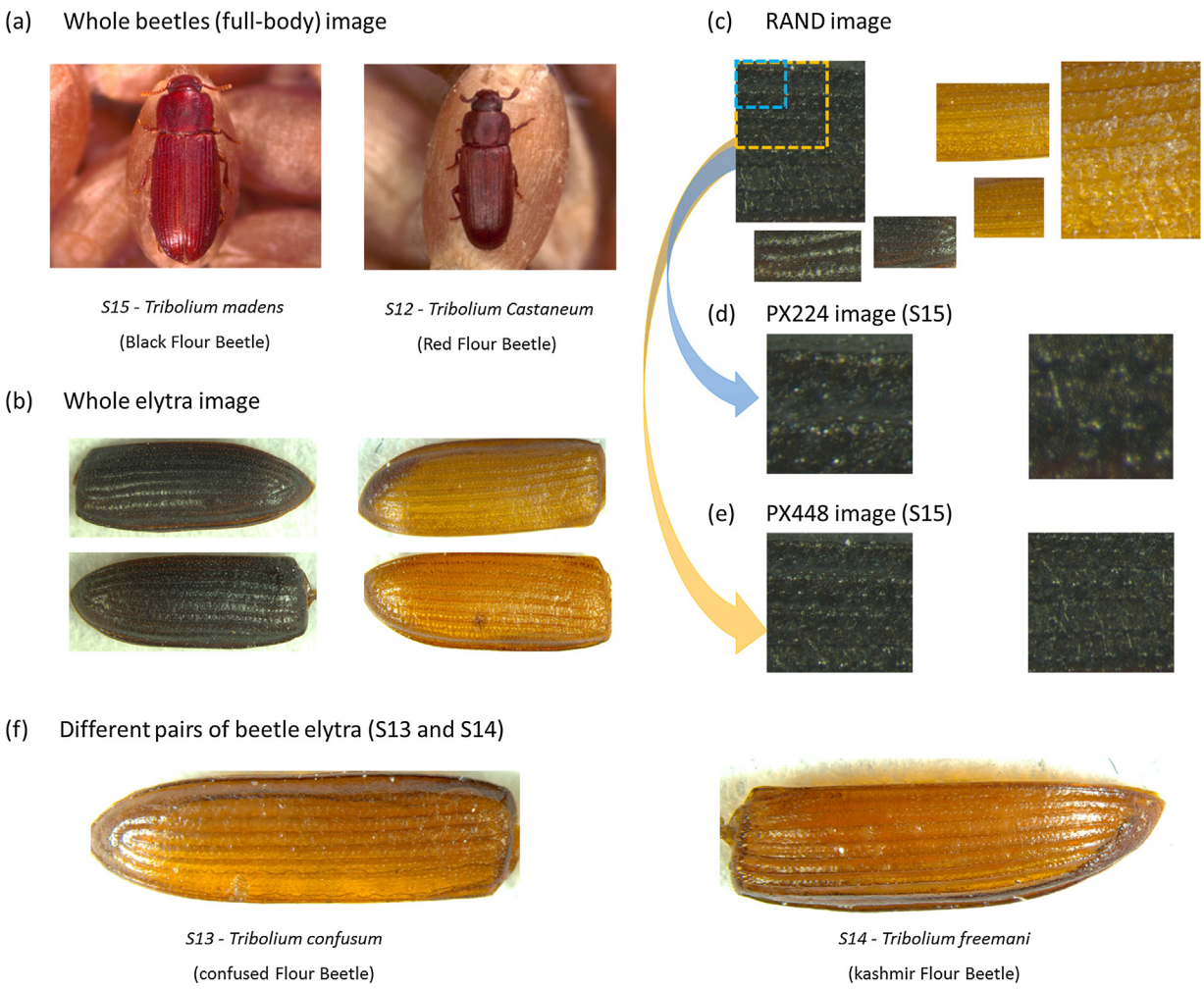

Fig. 1. . (a) S15 - *Tribolium madens* (Black Flour Beetle), on the left - and S12 - *Tribolium Castaneum* (Red Flour Beetle), on the right. (Source: https://www.grainscanada.gc.ca; access date: 07/01/2019) (b) whole elytra images capture via lab camera; (c) RAND dataset; random fragmented elytra images for S12 and S15; (d) and (e) PX224 and PX448 dataset; fragmented elytra image with fixed cropping size, which covered diffe rent size of the elytra. (f) one example of difficult pairs of beetles (S13 and S14).

food samples, we generated 100 randomly cropped images for each RAW image (i.e. whole elytra image) in RGB format. In details, the fragmented images were randomly cropped with width and height varied from 400 to 4000 in pixels, and no background would be captured in the fragmented image. Notably, this dataset is the same image dataset already adopted in previous published research studies, (Park et al., 2016; Bisgin et al., 2018), to allow for a fair comparison of classification performance. This dataset will be further referred to as RAND since the images in this dataset have different (and random) width and height. Notably, since our VGG16-based model requires a fixed input size of 224 pixels in height and width (hereafter 224-by-224 for short), each RAND image is resized to fit this requirement. In addition, to investigate the effect of fixed image resolutions to classification performance, we generated two others dataset out of the RAND dataset, namely PX224 and PX448. To do so, each image in RAND dataset is cropped into multiple sub-images (i.e. one sub-image at each corner and one in the central of the images) at the considered fixed size. Thus, we generated 5 fixed size images for each RAND image, leading to a total of 500 fixed size images for each original RAW image (whole elytra image). The PX224 and PX448 datasets have been generated by considering a fixed cropping size of 224-by-224 and 448-by-448 pixels, respectively.

Although the final input size of all these datasets (RAND, PX224 and PX448) in the network model is the same, the covered elytra region of these dataset is significantly different since we used a fixed magnification ratio in the lab camera to capture the elytra image. For example, PX448 image will cover 4 times larger region of elytra than PX224 image. Although larger size image may contain more features of the beetle species, it may also lose information during resolution reduction in the resizing process to fit VGG16 requirements.

2.3. Convolutional neural network

Convolutional neural networks (CNNs) are a class of deep neural architectures widely adopted in computer vision for object classification and automatic segmentation (Krizhevsky et al., 2012; Lawrence et al., 1997; LeCun et al., 1998). Different from other image classification algorithms (Zheng et al., 2006), CNNs need little pre-processing on the image, as they learn the features (i.e., weights of the convolutional filters) directly during the network training process. In details, the architecture of a typical CNN is composed of a sequence of convolutional and pooling layers, terminating with a block of fully-connected layers. Each convolutional block is constituted by one or many convolutional filters. Each filter will apply a convolution operation to the input matrix and generate an output activation map that will be fed as the input to the next layer. Pooling layers are used to apply subsampling, usually via maximum or average value of neighborhood pixels. Finally, a block of dense layers is composed of a series of fully

connected neural network layers aimed at combining the features resulting from previous convolutional layers. Frequently a CNN architecture may also include other operations such as the dropout, used to avoid overfitting and to help the network better generalize, or the Batch Normalization (Ioffe and Szegedy, 2015), introduced with the purpose of normalizing the variability in the data batches.

Selecting the network structure to best match the classification task is crucial for accuracy and good generalization on unseen data. The choice of the structure is currently based on semi-automated optimization procedures applied to main types of architecture and variants of the CNNs. In particular, LeNet (LeCun, 2015), VGG (Simonyan and Zisserman, 2014), and Inception network (Szegedy et al., 2015), have been widely used in the deep learning community for different tasks. With 138 million of parameters, VGG16 is one of the best-performing convolution based structures trained and benchmarked on the ImageNet dataset (Krizhevsky et al., 2012) and it is the one that has been used in this study.

2.4. Network model structure

Due to the limited number of available beetle images, it is problematic to train a large deep network model from scratch. Therefore, we used a pre-trained network model, and a transfer learning approach (Pratt, 1993) to adapt the model to our learning setting. To do so, the pre-trained weights of the 5 convolutional blocks in the VGG16 architecture were frozen, thus allowing only the final dense block to receive gradient updates during the training epochs. In our model, the dense block was composed of 4 fully connected (FC) hidden layers. The number of nodes in hidden layers were 1024, 512, 256 and 128 respectively. One dropout unit was added after the first FC layer (1024 nodes) with a drop rate equal to 0.8. We considered ReLU (Rectified Linear Units) as the activation function for each FC layer (Nair and Hinton, 2010). The dense block included a last output layer with Softmax activation for the classification of the 15 beetle species (Fig. 2). The network weights in the FC layers were trained via back propagation for 1000 epochs. To balance the vectorization ability with the total sample size, we also considered a mini-batch setting with a batch size = 128 in each epoch for each training and modeling process.

2.5. Network optimizers

We evaluated four different optimization strategies, stochastic gradient descent (SGD), RMSprop (Tieleman and Hinton, 2012), Adam (Kingma and Ba, 2014) and Adadelta (Zeiler, 2012)) with Categorical Cross-entropy loss function. For SGD, we empirically noted that the default value of 10e-2 for the learning rate was too high, and so we tuned it to 10e-3. For other optimizers we used default hyper-parameters.

2.6. Cross validation

Non-exhaustive “Leave-many-out” cross validation (also called repeated learning-testing, RLT) (Arlot and Celisse, 2010) was used to evaluate the model performance. For each time, the whole dataset would be randomly split into a train set including 5400 images, and a test set including the remaining 1500 images. The cross-validation process was repeated 100 times.

Since augmented images from the same raw image may include overlapped areas; treating them as independent samples in training-testing split will cause information leak and overfitting. The 100 sub-images from the same raw image were grouped together when assigned to training or testing dataset.

2.7. Code and experimental environment

Keras (version 2.0.6) (Chollet, 2015) and TensorFlow (version 1.1.0) (Abadi et al., 2016) were used for network construction. The working environment was based on CentOS 6.9 and Python 3.6.1, on an in-house High Performance Computing Cluster with ~1100 CPU cores. The script for modeling used in this study is available on github (https://github.com/seldas/DL Beetles).

3. Results

3.1. Framework overview

The overview workflow of the deep learning framework developed in this study is displayed in Fig. 3. Three datasets (RAND, PX224 and PX448) were prepared for the analysis (see Materials and Methods). The network model was constructed with two main parts: P1 (convolutional layers) and P2 (fully connected layers). In P1 where transfer learning was applied, all network weights were imported from pre-trained network on ImageNet and fixed during our own training process. The output feature matrix of P1 was then passed to P2 as input. Leave-many-out cross validation was applied to systematically evaluate the network model performance, where for each run, the training and testing dataset would be randomly assigned.

The fully connected layers in P2 were trained via back propagation.

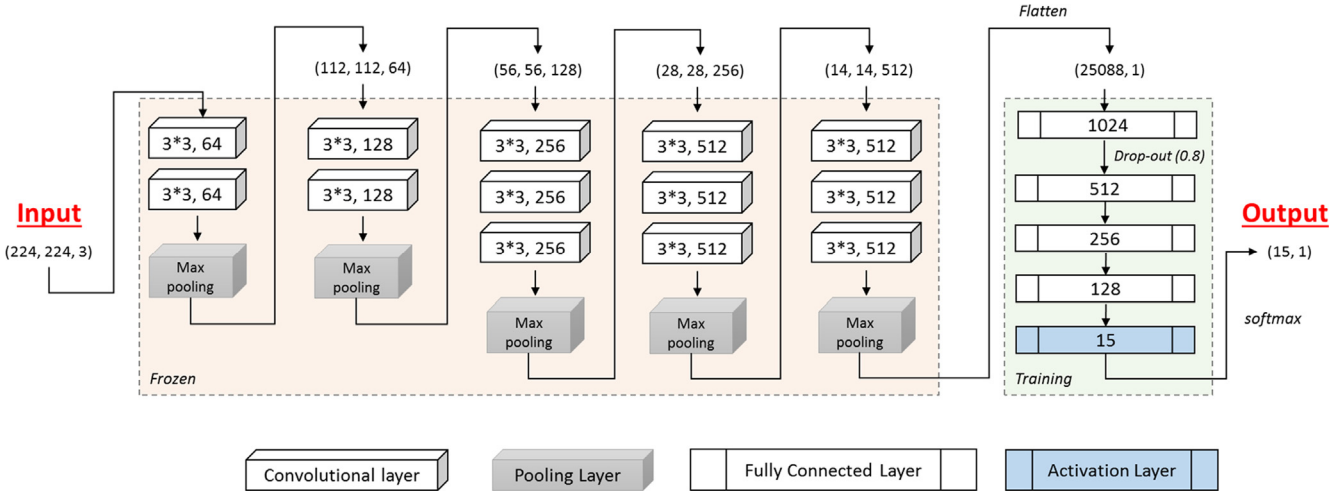

Fig. 2. The customized VGG16 Network structure for the beetle identification task (15 classes). Starting from a VGG16 architecture, the weights in convolutional layers are kept frozen during training, while the final fully connected layers are customized and trained for 1000 epochs.

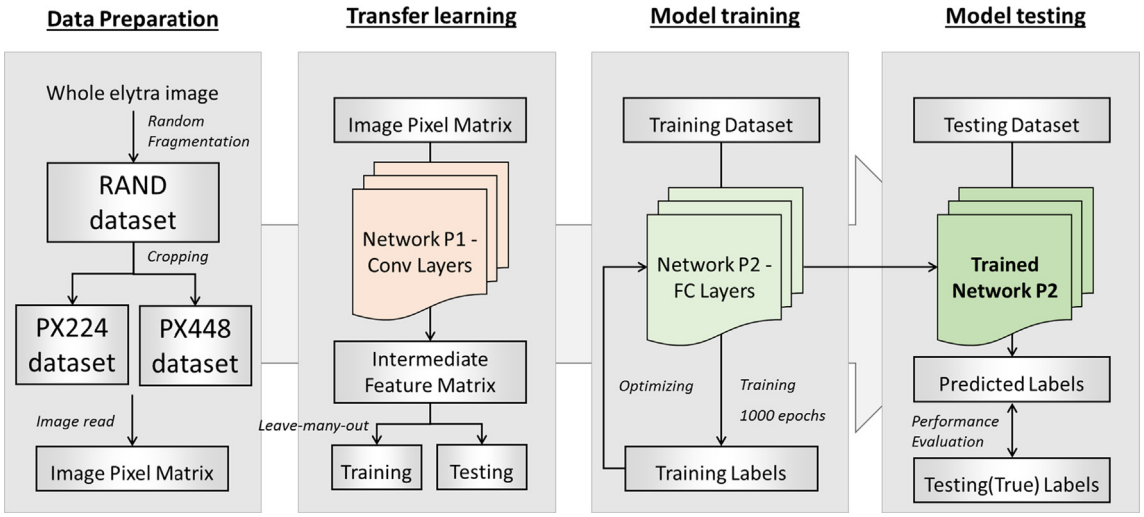

Fig. 3. Overview of beetle species recognition pipeline based on fragmented elytra image.

Weights in network P2 were updated within 1000 epochs. The model was then used to predict testing images, which were not used in model training process, for performance evaluation.

3.2. Model training and optimization

In our study, we investigated the best cropping size of input image, and the best optimizer in combination with our VGG16-based model. Three datasets with different cropping sizes (PX224, PX448 and RAND) and four different optimizers (SGD, RMSprop, Adam, Adadelta) have been considered. All together, we experimented 12 deep learning models with different combinations of cropping size and optimizer. For each combination, we repeated 5 times in RLT to evaluate its performance.

The average accuracy of prediction over the 5 repetitions for all the tested combinations is shown in Table 2, along with corresponding 95% Bootstrap Confidence Intervals. Overall, the SGD optimizer with PX448 resulted in the best combination for both training and testing accuracy. Although RAND and PX448 have similar performance, performance obtained on the PX224 dataset are not as good as the ones on the other two datasets regardless the optimizer used. The poor performance of PX224 may be explained as considered images may not contain enough significant patterns allowing the model to classify correctly the species. Considering the different optimizers, SGD and Adadelta resulted on average in the better performance, with respect to the other two, RMSprop and Adam. In Table 2 we highlighted in bold the combinations of parameters leading to the best performance.

Table 2
Performance of DL model on three datasets with four optimizers with 5 repeats.

Optimizer	Dataset	Train Accuracy (Min CI, Max CI)	Cross Validation Accuracy (Min CI, Max CI)
SGD	RAND	0.995 (0.995, 0.996)	0.814 (0.783, 0.813)
	PX448	**0.997 (0.996, 0.997)**	**0.848 (0.814, 0.874)**
	PX224	0.969 (0.966, 0.972)	0.698 (0.703, 0.724)
RMSprop	RAND	0.941 (0.937, 0.948)	0.659 (0.618, 0.68)
	PX448	0.661 (0.629, 0.761)	0.518 (0.449, 0.592)
	PX224	0.779 (0.749, 0.844)	0.414 (0.357, 0.459)
Adam	RAND	0.926 (0.917, 0.934)	0.53 (0.48, 0.563)
	PX448	0.955 (0.949, 0.963)	0.391 (0.358, 0.471)
	PX224	0.85 (0.837, 0.86)	0.107 (0.069, 0.158)
Adadelta	RAND	0.995 (0.994, 0.996)	0.833 (0.803, 0.847)
	PX448	0.996 (0.996, 0.997)	0.815 (0.779, 0.852)
	PX224	0.975 (0.972, 0.977)	0.583 (0.588, 0.623)

Furthermore, we also investigated the training process over 1000 epochs. The trends of training loss (i.e., Categorical Cross-entropy) over 1000 epochs for all four optimizers are shown in Fig. 4a. We observed a lower loss score of SGD and Adadelta, with Adadelta already converging to its minimum after 400 epochs. However, the loss score of SGD is generally better. On the other hand, the loss score of RMSprop is increasing after ~100 epochs, which means the optimizing process is not working well to find the optimum point. Since SGD and Adadelta are already working well for our dataset, we did not further tune RMSprop hyper-parameters. The trends of training process for all three datasets have been also investigated (Fig. 4b). Note that the performance of PX448 with SGD optimizer occurred in both figures can be used as a baseline. In general, PX448 performs best among all three datasets. However, since the Y-axis is log10-based, the performance for all three datasets is acceptable as final loss score (Categorical Cross Entropy) is lower than 0.1, which is much better than those models using Adam or RMSprop optimizers. We then evaluated both training and testing performance of SGD + PX448, the best model based on our training performance. As shown in Fig. 4c, although the training loss kept reducing, the testing loss and accuracy saturated after 100 epochs, which indicated the model quickly reached its optimum point.

In summary, we demonstrated PX448 is the best input size for our model. Compared to RAND, PX448 would increase the model applicability to handle smaller fragments, and PX224 did not hold a satisfying performance based on the result. In terms of optimization, we think either SGD or Adadelta works adequately, as they showed similar modeling performance. We further demonstrated the model performance is reliable by evaluating with 5 repetition, as we extended the RLT repeats from 5 to 100 times for two best models, PX448 with SGD and PX448 with Adadelta. The averaged categorical accuracy among 100 repeats for PX448 with SGD and Adadelta was 0.838 (CI: 0.829, 0.846) and 0.816 (CI: 0.808, 0.825) respectively, which are concordant to these evaluated by 5 repetition.

3.3. Model performance per species

We selected the PX448 input size with SGD optimizer as it held the best performance in both training and testing performance for performance per species evaluation. In details, the precision and recall of each of 15 beetle species are shown in Fig. 5. Generally, five species (S01, S03, S07, S11 and S15) showed good predicting performance, and maintained a high and steady score in both precision and recall during 100 repeats. Three species (S02, S09 and S14) showed a relatively low precision (median value < 0.75); three species (S05, S13 and S14) had relatively high deviation of precision (large hinges, whiskers and

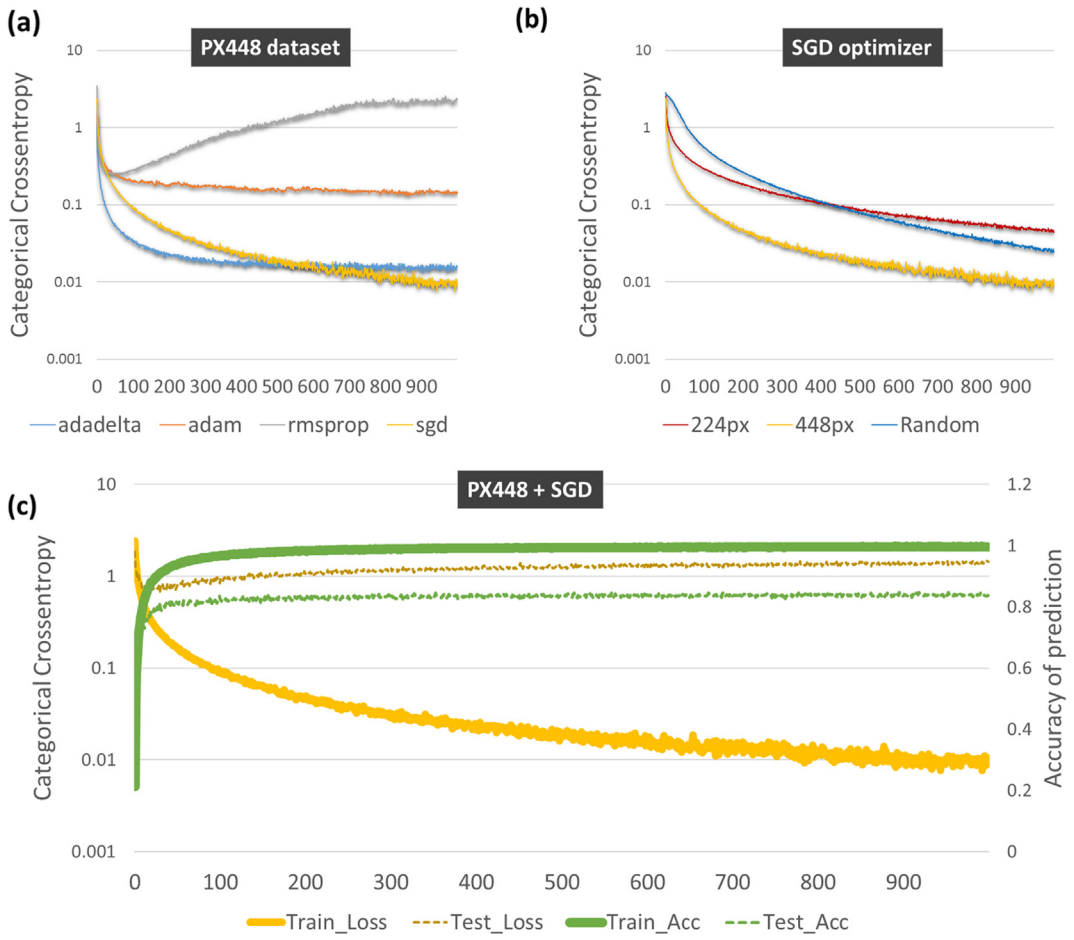

Fig. 4. Tuning deep learning model. (a) training process of four optimizers with PX448 dataset; (b) training process of three datasets with SGD optimizer. The X-axis is the training epochs from 0 to 1000; Y-axis is the loss score i.e., categorical crossentropy. The value is averaged of 5 repeats. (c) Training and testing performance of SGD + PX448. Primary Y-axis (left) is for loss score, secondary y-axis (right) is for accuracy of prediction. Yellow and green lines are for loss score and accuracy, where bold and thin lines are for training and testing, respectively.

outliers); two species (S05 and S10) showed moderate low recall (< 0.75). The overall performance for each species was measured by F1 score, that is the harmonic mean of Precision and Recall. As reported in Fig. 5, species S05 and S10 showed relatively weak predicting performance among all 15 species; S08 and S13 had the most lower end outliers, which imply these beetle species are difficult to be distinguished by the model.

4. Discussion

In this study, we adopted a deep learning network model to recognize beetle species that are usually found in food products. To obtain the optimized network model, we investigated three different input image sizes and tuned different optimizers used in model training. The best model with PX448 and SGD optimizer, yielded an averaged 0.838 accuracy of prediction in testing dataset, which is competitive to our previous machine learning models using ANN (0.79) or SVM (0.85) (Bisgin et al., 2018), which have also been optimized after extensive feature selection and parameter optimization.

The advantage of the DL algorithm is not only having a competitive performance, which is expected if the dataset is large enough; but the automatic feature design will also make deep learning more attractive, as researchers do not need to human-engineered image features, a very time and labor-consuming step where the expert skill is essential. For example, in our previous studies using ANN and SVM, we manually engineered the image features including 210 global features (such as color, shape, etc.) and 415 local features (such as edges, lines, etc.). Those features were designed to capture patterns like hairs, holes, lines, etc., which used the most of labor and time during the whole study. Based on our previous experience, good features that capture the most important patterns of the image are the most impacting factor and often dominate the model performance regardless of the model type. By using deep learning, since input is the image themselves, theoretically a well-trained deep learning model will not lose any pattern information that exists in the original image. Although DL models will automatically design and fine-tune features during the learning process, there are usually millions of weights needs to be fine-tuned and the learning process may be quite long to get a good performance. Therefore, appropriate selection of network architecture and its hyper-parameters would be essential for achieving a well-performed model. There are a lot of hyper-parameters that can be tuned; for example, different network architecture like AlexNet, VGG, or ResNet may perform differently on the dataset; even with the same network architecture, different initialization and optimization, and activation method would also affect the model training time and final performance. During our research, we observe that the performance of SGD would be better on RAND dataset if the learning rate parameter is set to 10e-3 rather than the default value (10e-2); dropout works better than Batch Normalization as a regularization method. Since the focus of this paper majorly relies on the applicability of deep learning, we don't include in the paper the results of all hyper-parameter optimization process.

On the other hand, a good initialization of the network weights, or pre-trained networks may significantly reduce the time required for the training process. As we demonstrated in our study, transfer learning by

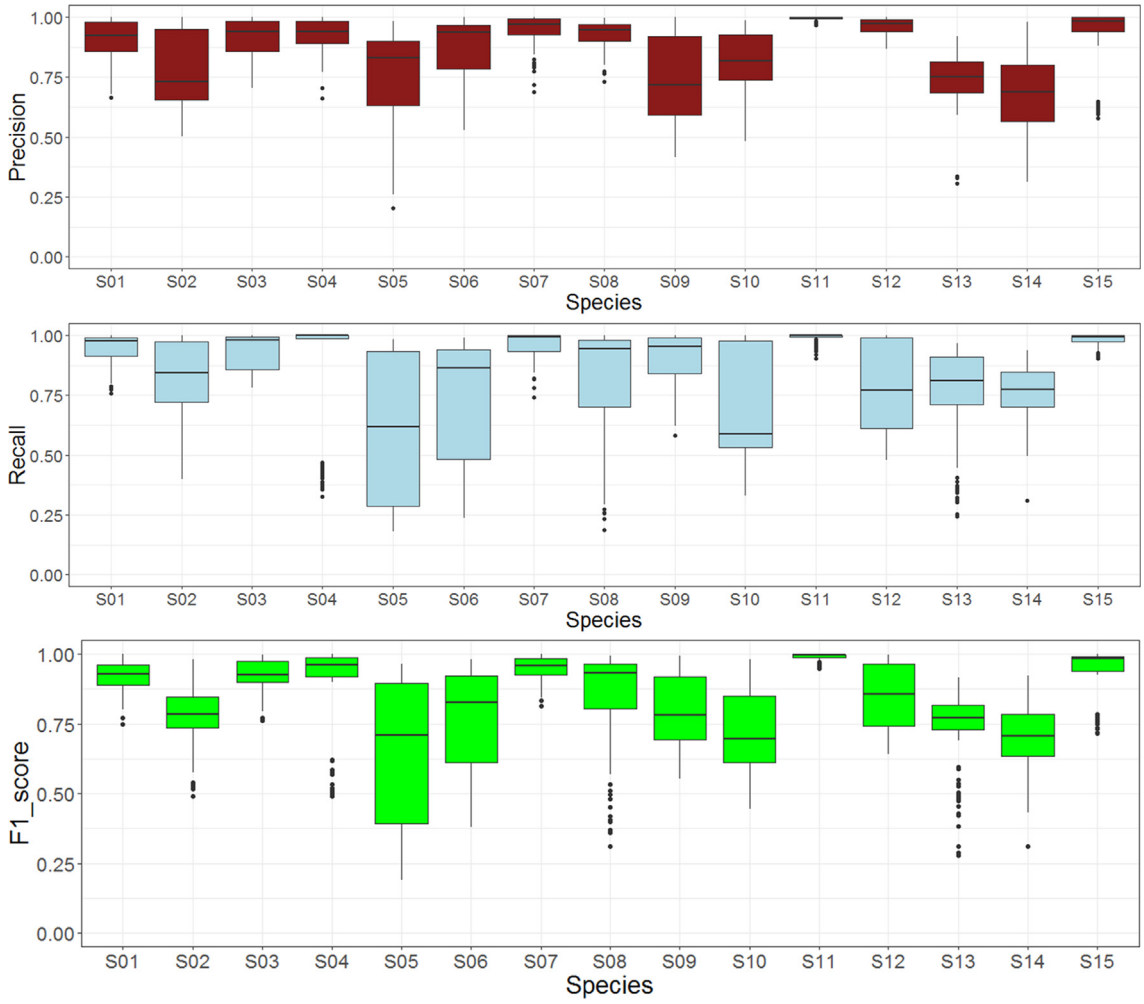

Fig. 5. Overall precision (red), recall (blue) and F1 (green) scores per species in 100 RLTs. The Tukey box and whiskers plot reveals the median (base line in the box), the first and third quartiles of the data (hinge of box) and the 1.5 inter-quartile range (IQR) from the hinge (upper and lower whiskers). Outliers are marked in black dots. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

using pre-trained network weights can be helpful to this specific task of beetle species recognition based on elytra fragments.

Although we have demonstrated that PX448 with SGD will hold the best performance for current deep learning model, there is still a lot of room for improvement that can be done in the future. One important factor to model performance is the training data size. As known, deep learning model performs better especially on large dataset. With the accumulation of beetle elytra images, we expect our model can be more accurate and may generalize to recognize more beetle species. Moreover, a large dataset will increase the flexibility of the training process. As currently we freeze the whole convolutional part of the model due to lack of images, we may also consider extending gradient updates also to the last one or two convolutional blocks if we collect enough images, and so giving the network the ability to fine-tune its own set of learnt features on the beetle elytra images. Also, with the fast development of new network architectures, other network structures such like ResNet, AlexNet and GoogLeNet may achieve a better performance than VGG.

Our current model is training and fine-tuning on in-silico fragmented beetle elytra images, although it is also captured by the same lab camera, it still differs from the real sample found in the food remain. One of the most important differences between the in-silico mimic and real fragments is that the texture may change after food-processing. There are two concerns that we decided to use mimic fragments: first is because of the sample size and data augmentation. It is quite time and labor-consuming to go through the whole imaging process to obtain one qualified image for both fragments and whole elytra, that is also the main reason why only 69 entire beetle images were obtained in the dataset. Since the real fragments are usually very small and differs in shapes, one real fragment may not be applicable to use due to its shape, and it is difficult to do data augmentation. Meanwhile, it is possible to crop multiple different fragments from a whole elytra image, and it is easy to get fragment images with good shapes. Another concern is about the ground truth of the dataset. As known, labeling preparation for the dataset used in deep learning is always the most difficult part, and it is even harder for beetle elytra fragments since many of them are difficult to recognize even for an experienced expert. Meanwhile, it is much easier to label the whole elytra and, in such case, provide a more reliable endpoint for model training.

Additionally, since in most cases transfer learning will perform better when the source and target dataset are similar, it is thus important to use a pre-trained network that is optimized by similar objects. To the best of our knowledge, we are not aware of any beetle fragment model that has been trained publicly, and our model is the first one that is optimized based on fragment images. We believe it is much similar to the scenario of real fragments than other pre-trained models that are trained for common objects recognition. Meanwhile, since the in-silico fragmented image is relatively easier to obtain and assign species labels, it is also easier to optimize when more data is retrieved. The further optimized model would have a chance to perform

better when transferred to a real fragment recognition model, and further research and investigations are essentially needed.

5. Conclusion

This study has demonstrated that deep learning has a promising potential in assisting laboratory processes need for food safety regulation. With the advantages of automated feature design and transfer learning, deep learning models will help FDA investigators quickly and accurately recognize the species of pest in food contamination, and it also holds the potential to expand to other filth elements in food contamination, not only elytra fragments. We acknowledge that the model performance may be inflated since the current dataset used in this study is small. To further improve the model performance and applicability, processing more beetle images to enlarge the dataset is necessary.

Disclaimer

The views presented in this article do not necessarily reflect the current or future opinion or policy of the U.S. Food and Drug Administration. Any mention of commercial products is for clarification and not intended as an endorsement.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., 2016. TensorFlow: a system for large-scale machine learning. OSDI 265–283.

Arlot, S., Celisse, A., 2010. A survey of cross-validation procedures for model selection. Statistics Surveys 4, 40–79.

Bansal, S., Singh, A., Mangal, M., Mangal, A.K., Kumar, S., 2017. Food adulteration: Sources, health risks, and detection methods. Crit. Rev. Food Sci. Nutr. 57, 1174–1189.

Bengio, Y., 2012. Deep learning of representations for unsupervised and transfer learning. In: Proceedings of ICML Workshop on Unsupervised and Transfer Learning, pp. 17–36.

Bisgin, H., Bera, T., Ding, H., Semey, H.G., Wu, L., Liu, Z., Barnes, A.E., Langley, D.A., Pava-Ripoll, M., Vyas, H.J., 2018. Comparing SVM and ANN based machine learning methods for species identification of food contaminating beetles. Sci. Rep. 8.

Boyer, D.M., Lipman, Y., Clair, E.S., Puente, J., Patel, B.A., Funkhouser, T., Jernvall, J., Daubechies, I., 2011. Algorithms to automatically quantify the geometric similarity of anatomical surfaces. Proceedings of the National Academy of Sciences.

Breiman, L., 2001. Random forests. Machine Learning 45, 5–32.

Cardinale, M., Luvisi, A., Meyer, J.B., Sabella, E., De Bellis, L., Cruz, A.C., Ampatzidis, Y., Cherubini, P., 2018. Specific Fluorescence in Situ Hybridization (FISH) Test to Highlight Colonization of Xylem Vessels by Xylella fastidiosa in Naturally Infected Olive Trees (Olea europaea L.). Front. Plant Sci. 9, 431.

Chollet, F., 2015. Keras.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Machine Learning 20, 273–297.

Cruz, A., Ampatzidis, Y., Pierro, R., Materazzi, A., Panattoni, A., De Bellis, L., Luvisi, A., 2019. Detection of grapevine yellows symptoms in Vitis vinifera L. with artificial intelligence. Comput. Electron. Agric. 157, 63–76.

Cruz, A.C., Luvisi, A., De Bellis, L., Ampatzidis, Y., 2017. X-FIDO: an effective application for detecting Olive Quick Decline Syndrome with deep learning and data fusion. Front. Plant Sci. 8, 1741.

Daly, H.V., Hoelmer, K., Norman, P., Allen, T., 1982. Computer-assisted measurement and identification of honey bees (Hymenoptera: Apidae). Ann. Entomol. Soc. Am. 75, 591–594.

DeChant, C., Wiesner-Hanks, T., Chen, S., Stewart, E.L., Yosinski, J., Gore, M.A., Nelson, R.J., Lipson, H., 2017. Automated identification of northern leaf blight-infected maize plants from field imagery using deep learning. Phytopathology 107, 1426–1432.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L., 2009. Imagenet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2009. IEEE, pp. 248–255.

Gaston, K.J., O'Neill, M.A., 2004. Automated species identification: why not? Philos. Trans. Royal Soc. London B: Biol. Sci. 359, 655–667.

Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y., 2016. Deep Learning. MIT press, Cambridge.

Gorham, J.R., 1979. The significance for human health of insects in food. Annu. Rev. Entomol. 24, 209–224.

Ioffe, S., Szegedy, C., 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167.

James F. Champbell Frank H. Arthur Michael A. Mullen Insect managment in food processing facilities S. Taylor Advances in Food and Nutrition Research 2004 Elsevier Academic Press London, UK 240 297.

Kagaya, H., Aizawa, K., 2015. Highly accurate food/non-food image classification based on a deep convolutional neural network. In: International Conference on Image Analysis and Processing. Springer, pp. 350–357.

Kagaya, H., Aizawa, K., Ogawa, M., 2014. Food detection and recognition using convolutional neural network. In: Proceedings of the 22nd ACM international conference on Multimedia. ACM, pp. 1085–1088.

Karunakaran, C., Jayas, D., White, N., 2004. Identification of wheat kernels damaged by the red flour beetle using X-ray images. Biosyst. Eng. 87, 267–274.

Kingma, D.P., Ba, J., 2014. Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Krizhevsky, A., Sutskever, I., Hinton, G.E., 2012. Imagenet classification with deep convolutional neural networks. Adv. Neural Inform. Process. Sys. 1097–1105.

Larios, N., Deng, H., Zhang, W., Sarpola, M., Yuen, J., Paasch, R., Moldenke, A., Lytle, D.A., Correa, S.R., Mortensen, E.N., 2008. Automated insect identification through concatenated histograms of local appearance features: feature vector generation and region detection for deformable objects. Mach. Vis. Appl. 19, 105–123.

Lawrence, S., Giles, C.L., Tsoi, A.C., Back, A.D., 1997. Face recognition: a convolutional neural-network approach. IEEE Trans. Neural Networks 8, 98–113.

LeCun, Y., 2015. LeNet-5, convolutional neural networks. URL: http://yann.lecun.com/exdb/lenet, 20.

LeCun, Y., Bengio, Y., Hinton, G., 2015. Deep learning. Nature 521, 436.

LeCun, Y., Bottou, L., Bengio, Y., Haffner, P., 1998. Gradient-based learning applied to document recognition. Proc. IEEE 86, 2278–2324.

Lee, S.H., Chan, C.S., Wilkin, P., Remagnino, P., 2015. Deep-plant: plant identification with convolutional neural networks. In: IEEE International Conference on Image Processing (ICIP). IEEE, pp. 452–456.

Lu, Y., Yi, S., Zeng, N., Liu, Y., Zhang, Y., 2017. Identification of rice diseases using deep convolutional neural networks. Neurocomputing 267, 378–384.

Nair, V., Hinton, G.E., 2010. Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp. 807–814.

Park, S.I., Bisgin, H., Ding, H., Semey, H.G., Langley, D.A., Tong, W., Xu, J., 2016. Species Identification of Food Contaminating Beetles by Recognizing Patterns in Microscopic Images of Elytra Fragments. PLoS ONE 11, e0157940.

Pratt, L.Y., 1993. Discriminability-based transfer between neural networks. Adv. Neural Inform. Process. Sys. 204–211.

Rees, D., 2004. Insects of Stored Products. CSIRO.

Reinholds, I., Bartkevics, V., Silvis, I.C., van Ruth, S.M., Esslinger, S., 2015. Analytical techniques combined with chemometrics for authentication and determination of contaminants in condiments: a review. J. Food Compos. Anal. 44, 56–72.

Schmidhuber, J., 2015. Deep learning in neural networks: an overview. Neural Networks 61, 85–117.

Simonyan, K., Zisserman, A., 2014. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

Stejskal, V., Hubert, J., Aulicky, R., Kucerova, Z., 2015. Overview of present and past and pest-associated risks in stored food and feed products: European perspective. J. Stored Prod. Res. 64, 122–132.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A., 2015. Going deeper with convolutions. In: Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition, pp. 1–9.

Tieleman, T., Hinton, G., 2012. Lecture 6.5-rmsprop: divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks Machine Learning 4, 26–31.

Weeks, P., Gauld, I., Gaston, K., O'Neill, M., 1997. Automating the identification of insects: a new solution to an old problem. Bull. Entomol. Res. 87, 203–211.

Wen, C., Wu, D., Hu, H., Pan, W., 2015. Pose estimation-dependent identification method for field moth images using deep learning architecture. Biosys. Eng. 136, 117–128.

Wu, Y., Trepanowski, N.F., Molongoski, J.J., Reagel, P.F., Lingafelter, S.W., Nadel, H., Myers, S.W., Ray, A.M., 2017. Identification of wood-boring beetles (Cerambycidae and Buprestidae) intercepted in trade-associated solid wood packaging material using DNA barcoding and morphology. Sci. Rep. 7, 40316.

Zeiler, M.D., 2012. ADADELTA: an adaptive learning rate method. arXiv preprint arXiv:1212.5701.

Zheng, C., Sun, D.-W., Zheng, L., 2006. Recent developments and applications of image features for food quality evaluation and inspection–a review. Trends Food Sci. Technol. 17, 642–655.

Zou, W., Chi, Z., Lo, K.C., 2008. Improvement of image classification using wavelet coefficients with structured-based neural network. Int. J. Neural Syst. 18, 195–205.

Zurek, L., Gorham, J.R., 2008. Insects as vectors of foodborne pathogens. Wiley Handbook of Science and Technology for Homeland Security.