Algorithms for Intelligent Systems
Series Editors: Jagdish Chand Bansal · Kusum Deep · Atulya K. Nagar

Mohammad Shorif Uddin
Jagdish Chand Bansal *Editors*

Computer Vision and Machine Learning in Agriculture

Springer

Algorithms for Intelligent Systems

This book series publishes research on the analysis and development of algorithms for intelligent systems with their applications to various real world problems. It covers research related to autonomous agents, multi-agent systems, behavioral modeling, reinforcement learning, game theory, mechanism design, machine learning, meta-heuristic search, optimization, planning and scheduling, artificial neural networks, evolutionary computation, swarm intelligence and other algorithms for intelligent systems.

The book series includes recent advancements, modification and applications of the artificial neural networks, evolutionary computation, swarm intelligence, artificial immune systems, fuzzy system, autonomous and multi agent systems, machine learning and other intelligent systems related areas. The material will be beneficial for the graduate students, post-graduate students as well as the researchers who want a broader view of advances in algorithms for intelligent systems. The contents will also be useful to the researchers from other fields who have no knowledge of the power of intelligent systems, e.g. the researchers in the field of bioinformatics, biochemists, mechanical and chemical engineers, economists, musicians and medical practitioners.

The series publishes monographs, edited volumes, advanced textbooks and selected proceedings.

More information about this series at http://www.springer.com/series/16171

Mohammad Shorif Uddin · Jagdish Chand Bansal
Editors

Computer Vision and Machine Learning in Agriculture

Springer

Editors
Mohammad Shorif Uddin
Department of Computer Science
and Engineering
Jahangirnagar University
Dhaka, Bangladesh

Jagdish Chand Bansal
Department of Applied Mathematics
South Asian University
New Delhi, India

ISSN 2524-7565 ISSN 2524-7573 (electronic)
Algorithms for Intelligent Systems
ISBN 978-981-33-6423-3 ISBN 978-981-33-6424-0 (eBook)
https://doi.org/10.1007/978-981-33-6424-0

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721, Singapore

Preface

In recent years, computer vision, a noncontact- as well as a nondestructive-technique involving the development of theoretical and algorithmic tools for automatic visual understanding and recognition finds huge applications in agricultural productions. Rendering the machine learning techniques to computer vision algorithms is boosting this sector with better productivity by developing more precise systems.

Computer vision and Machine learning (CV-ML) helps in plant disease assessment along with crop condition monitoring to control the degradation of yield, quality, and severe financial loss for farmers. Significant scientific and technological advances have been made in defect assessment, quality grading, disease recognition, pests, insects, fruits, and vegetable recognition and evaluation of a wide range of agricultural plants, crops, leaves, and fruits. Intelligent robots and drones developed with the touch of CV-ML can help farmers to perform various tasks like planting, weeding, harvesting, plant health monitoring, and so on.

This book is intended to publish the latest advances and developments in the field of computer vision, machine learning including deep learning tools and techniques in assessing and monitoring agricultural products and productions. The topics covered include plant, leaf, and fruit-disease detection, crop health monitoring, applications of robots and drones in agriculture, precision farming, assessment of product quality and defects, pest, insect, fruits, and vegetable types recognition, etc. It contains 11 chapters.

Chapter "Introduction to Computer Vision and Machine Learning Applications in Agriculture" highlights an introduction to computer vision and machine learning applications in agriculture. Chapter "Robots and Drones in Agriculture—A Survey" provides a comprehensive survey on several robotic applications in agriculture such as navigation, grafting, picking, weeding, spraying, harvesting, etc. It also illustrates the commercialization and challenges of real-fields applications of robots and drones in drones in boosting agricultural productions. Chapter "Detection of Rotten Fruits and Vegetables Using Deep Learning" describes a computer vision-based deep convolutional neural network for the detection of rotten fruits and vegetables. It performs experimentation with a dataset containing enough number of images of fresh and rotten fruits and confirms that the proposed deep learning architecture outperforms the existing approaches. Chapter Deep Learning-Based Essential Paddy Pests' Filtration Technique for Economic Damage Management" illustrates a region-based deep convolutional neural network known as Faster R-CNN to perform the detection and identification of both beneficial and non-beneficial paddy pests from the images. It has investigated three models of Faster R-CNN based on ResNet-101, VGG-16, and MobileNet and has obtained the highest accuracy from the ResNet-101-based Faster R-CNN. Besides, it developed an extensive dataset of paddy pests. Chapter "Deep CNN-Based Mango Insect Classification" discusses the creation of a quality dataset containing three different types of common mango insects. After that, it performs a classification of insects using an ensemble of three fine-tuned deep learning models, namely Xception, MobileNet, and VGG19. The ensemble model achieves a very good classification accuracy. Chapter "Implementation of a Convolutional Neural Network for the Detection of Tomato-Leaf Diseases" shows the implementation of a deep convolutional neural network for the early detection of tomato leaf diseases. Chapter "A Multi-Plant Disease Diagnosis Method Using Convolutional Neural Network" presents an optimal plant disease diagnosis model for multiple plants using convolutional neural networks. Chapter "A Deep Learning-Based Approach for Potato Disease Classification" investigates an early detection of potato disease through different deep CNN strategies by developing a dataset containing 7870 images of various diseases. Based on accuracy, precision, recall, and F1 score it finds that the ResNet is the best model for this particular application. Chapter "An In-Depth Analysis of Different Segmentation Techniques in Automated Local Fruit Disease Recognition" describes four different segmentation strategies, such as Otsu's method, *K-means* clustering, fuzzy *c*-means clustering, and region growing for the extraction of defective regions of the defective region of three common fruits of Bangladesh, namely guava, jackfruit, and papaya. *K-means* clustering technique gives the best performance among these segmentation techniques based on six quantitative analysis metrics by attaining an aggregate accuracy of 81.65%. Chapter "Machine Vision-Based Fruit and Vegetable Disease Recognition: A Review" presents a comprehensive survey of the recent advancement of computer vision and machine learning research efforts for fruit and vegetable disease recognition. It also shows a comparative study on these efforts based to find state-of-the-art techniques and shows ways for future research. Chapter "An Efficient Bag-of-Features for Diseased Plant Identification" proposes a bag-of-features based diseased plant identification method using gray relational analysis. It presents the experimental results on a publicly available leaf-image dataset (PlantVillage).

This book is expected to be very useful the researchers, academicians, undergraduate and postgraduate students who wish to work and explore the applications of computer vision and machine learning systems in the agricultural sector for boosting productions.

We sincerely appreciate the time, effort, and contribution of the authors and esteemed reviewers in maintaining the quality of the papers. Special thanks to the editors and supporting team of Springer for helping in publishing this book.

Dhaka, Bangladesh — Mohammad Shorif Uddin
New Delhi, India — Jagdish Chand Bansal

Contents

Editors and Contributors

About the Editors

Prof. Mohammad Shorif Uddin received his Ph.D. degree in Information Science from Kyoto Institute of Technology in 2002, Japan, Master of Technology Education degree from Shiga University, Japan, in 1999, Bachelor of Electrical and Electronic Engineering degree from Bangladesh University of Engineering and Technology in 1991, and also MBA from Jahangirnagar University in 2013. He began his teaching career as Lecturer in 1991 at Bangladesh Institute of Technology, Chittagong (Renamed as CUET). He joined the Department of Computer Science and Engineering of Jahangirnagar University in 1992, and currently, he is Professor of this department. In addition, he is Teacher-in-Charge of ICT Cell of Jahangirnagar University. He served as Chairman of Computer Science and Engineering of Jahangirnagar University from June 2014 to June 2017. He undertook postdoctoral researches at Bioinformatics Institute, Singapore, Toyota Technological Institute, Japan, Kyoto Institute of Technology, Japan, Chiba University, Japan, Bonn University, Germany, and Institute of Automation, Chinese Academy of Sciences, China. His research is motivated by applications in the fields of artificial intelligence, imaging informatics, and computer vision. Mohammad Uddin is IEEE Senior Member and Fellow of Bangladesh Computer Society (BCS) and The Institution of Engineers Bangladesh (IEB). He has lectured a good number of undergraduate and graduate courses, wrote more than 140 journal and conference papers, and organized some national and international conferences and seminars. He had delivered a remarkable number of keynotes and invited talks and acted as General Chair as well as TPC Chair of many international conferences. He holds two patents for his scientific inventions and received the Best Paper Award in the International Conference on Informatics, Electronics & Vision (ICIEV2013), Dhaka, Bangladesh, and Best Presenter Award from the International Conference on Computer Vision and Graphics (ICCVG 2004), Warsaw, Poland. He is an Associate Editor of IEEE Access.

Dr. Jagdish Chand Bansal is Associate Professor at South Asian University New Delhi and Visiting Faculty at Maths and Computer Science, Liverpool Hope University UK. Dr. Bansal has obtained his Ph.D. in Mathematics from IIT Roorkee. Before joining SAU New Delhi, he has worked as Assistant Professor at ABV-Indian Institute of Information Technology and Management Gwalior and BITS Pilani. His primary area of interest is swarm intelligence and nature-inspired optimization techniques. Recently, he proposed a fission–fusion social structure-based optimization algorithm, Spider Monkey Optimization (SMO), which are being applied to various problems from engineering domain. He has published more than 70 research papers in various international journals/conferences. He is Series Editor of the book series Algorithms for Intelligent Systems (AIS) and Studies in Autonomic, Data-driven and Industrial Computing (SADIC) published by Springer. He is Editor-in-Chief of the International Journal of Swarm Intelligence (IJSI) published by Inderscience. He is also Associate Editor of IEEE ACESSS published by IEEE and ARRAY published by Elsevier. He is Steering Committee Member and General Chair of the annual conference series SocProS. He is General Secretary of Soft Computing Research Society (SCRS). He has also received Gold Medal at UG and PG levels.

Contributors

Shovon Ahammed Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Farruk Ahmed Department of Computer Science and Engineering, Independent University, Dhaka, Bangladesh

Kawser Ahmed Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Morium Akter Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh

Md. Ariful Islam Arif Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Md. Abdul Aziz Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Jagdish Chand Bansal Department of Applied Mathematics, South Asian University, New Delhi, India

Rabeya Basri Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh

Md. Tarek Habib Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh;

Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Mohammad Reduanul Haque Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Mahedi Hasan Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Md. Mahmodul Hasan Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail, Bangladesh

Md. Zahid Hasan Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Mehedi Hasan Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail, Bangladesh

A. K. M. Kamrul Islam Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail, Bangladesh;
Department of Computer Science, Georgia State University, Atlanta, GA, USA

Fahmida Islam Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh

Md. Tarequl Islam Department of Information and Communication Technology, ICT Division, Dhaka, Bangladesh

Md. Ismail Jabiullah Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Susovan Jana School of Education Technology, Jadavpur University, Kolkata, India

Muhammad Mohsin Kabir Bangladesh University of Business and Technology, Dhaka, Bangladesh

Amina Khatun Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh

Himanshu Mittal Jaypee Institute of Information Technology, Noida, India

M. F. Mridha Bangladesh University of Business and Technology, Dhaka, Bangladesh

Abu Quwsar Ohi Bangladesh University of Business and Technology, Dhaka, Bangladesh

Raju Pal Jaypee Institute of Information Technology, Noida, India

Avinash Pandey Jaypee Institute of Information Technology, Noida, India

Ranjan Parekh School of Education Technology, Jadavpur University, Kolkata, India

Muntaka Rahat Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail, Bangladesh

Md. Syfur Rahman Education Management Information System (EMIS) Cell, Ministry of Education, Dhaka, Bangladesh

Mohammad Motiur Rahman Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail, Bangladesh

Mostafizur Rahman Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Mukesh Saraswat Jaypee Institute of Information Technology, Noida, India

Bijan Sarkar Department of Production Engineering, Jadavpur University, Kolkata, India

Sumaita Binte Shorif Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh

Mohammad Shorif Uddin Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh

Nusrat Zahan Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Nahid Zeba Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh

Introduction to Computer Vision and Machine Learning Applications in Agriculture

Mohammad Shorif Uddin and Jagdish Chand Bansal

1 Introduction

Computer vision and machine learning (CVML) techniques in agriculture play a major role to meet the increasing demand for food production [1, 2]. Population growth with rapid urbanization has made a drastic hike in food demand [3] including effective food production rate with a limited amount of cultivable lands and other resources in a sustainable way for precision agriculture [4, 5]. In recent years, the usage of computer vision techniques using expert machine learning algorithms in agriculture has significantly increased to enhance the efficiency of crop productivity in a more cost-effective manner [6, 7]. Research on CVML in agriculture has grabbed the attention for increased food production with minimum production costs due to the replacement of the traditional labor-intensive time-consuming works through the machine-based expert systems [8, 9]. Machine learning algorithms use Graphics Processing Units (GPUs) to analyze a massive amount of complex data to develop computational models with a high parallel processing capacity [10]. Thus, CVML techniques in the GPU environment have been widely used to perform diverse agricultural tasks by processing numerous images captured by humans, robots, drones, and other remote sensors to attain agricultural sustainability [11, 12]. On the contrary, computer vision approaches with deep convolutional neural network (CNN) have been increased in high-level scene understanding due to their intensive accuracy in feature representations [13–15] instead of traditional handcrafted classification

M. S. Uddin (✉)
Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh
e-mail: shorifuddin@gamil.com

J. C. Bansal
Department of Applied Mathematics, South Asian University, New Delhi, India
e-mail: jcbansal@gmail.com

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_1

of images [16]. This chapter presents an introduction to the CVML technologies in agriculture with current challenges and explores further scopes to develop an efficient reference for researchers.

The rest of the chapter is organized as follows: an introduction to CVML technologies in agricultural automation is presented in Sect. 2. Section 3 discusses the current challenges and future endeavors of CVML applications in agriculture. Finally, Sect. 4 provides a conclusion.

2 Computer Vision and Machine Learning in Agriculture

Computer vision is a high-level image processing that can automatically perform the diverse tasks of automatic detection, recognition, classification, monitoring of objects by analyzing the acquired images. A simple diagram of the computer vision flow process is shown in Fig. 1. Different machine learning strategies have emerged that can be easily combined with the computer vision process and gives better accuracy. Recently, deep learning strategies, such as CNN/RNN (convolutional neural network/recurrent neural network), have been widely used along with a vision-based

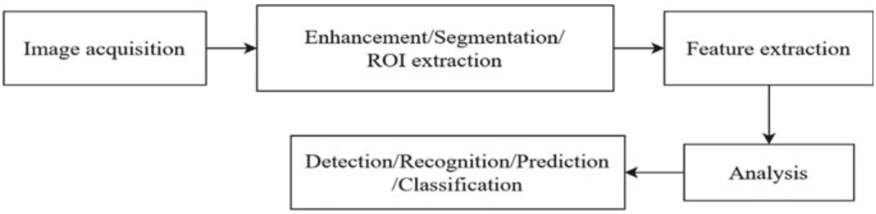


Fig. 1 A schematic flow diagram of the simple computer vision-based system

Fig. 2 A schematic flow diagram of computer vision-based machine learning as well as a deep learning system

Table 1 Shows the different machine-learning and deep learning approaches used in agricultural applications

Traditional machine learning algorithms	Deep learning algorithms
• Bayesian networks • Logistic regression • Decision trees • Bagging • Boosting • Random forests • K-nearest neighbor • Support vector machine (SVM) • Artificial neural network (ANN) • Genetic algorithm (GA) • Principal component analysis (PCA) • K-means clustering	• Multilayer Perceptron Neural Network (MLPNN) • Convolutional Neural Network (CNN) • Recurrent Neural Network (RNN) • Long Short-Term Memory Network (LSTM) • Generative Adversarial Network (GAN) • Deep Belief Network (DBN) • Recursive Neural Network • Region-based CNN (R-CNN), Fast R-CNN, Faster R-CNN

system that gives an extraordinary improvement of accuracy. Figure 2 shows a flow diagram of machine learning/deep learning along with a vision-based process.

In Table 1, we have mentioned the different machine-learning and deep learning approaches used in agricultural applications. As deep CNN is now, a de facto tool for diverse prediction, detection, recognition, classification, and clustering tasks; hence, we have summarized the different CNN models used in agricultural applications in Table 2.

Computer vision with machine learning has been applied in agriculture for more efficient and faster crop production to minimize the production cost and labor effort [17]. The applications of CVML in agriculture are summarized in Table 3.

A brief description of CVML applications is described below.

(a) Crop Growth Monitoring
Crop growth monitoring is one of the significant aspects of precision agriculture that uses a camera to capture images at different growth stages to improve

Table 2 Shows the different CNN models used in agricultural applications

Benchmark Deep CNN Models
• LeNet • VGGNet (VGG 16, VGG 19) • ResNet (ResNet 50, ResNeXt 50) • DenseNet (DenseNet 121, DenseNet 161, DenseNet 201) • GoogleNet (Inception V1, Inception V3, Inception V4) • Xception • AlexNet • MobileNet • NASNetMobile

Table 3 Computer vision and machine learning (CVML) application tasks in agriculture

Technique	Application
Computer vision and machine learning	Crop growth monitoring
	Disease, pest, and weed detection
	Automatic crop harvesting
	Product inspection and quality testing
	Plant phenotyping
	Species recognition
	Yield prediction
	Water management
	Soil management

production efficiency by reducing monitoring time and labor work. CVML techniques can detect the delicate changes in crop growth due to malnutrition at an early stage and efficiently monitoring crop health regularly [18–20].

(b) Diseases, Weeds, and Insects Detection
CVML uses diverse techniques in different types of crop diseases, pest, and weed detection [21–28]. An overall review of the use of vision-based systems in pests, diseases, and weeds detection was presented in [29].

(c) Automatic Crop Harvesting
CVML has brought revolutionary changes in the automation of different types of vegetables and fruits harvesting such as cucumber, apple, cherries, etc. using robotic systems [30–32].

(d) Product Inspection and Quality Testing
Several CVML techniques have been applied for inspection and quality testing of agricultural products particularly for fruits and vegetables and are described in [33–35].

(e) Plant Phenotyping
Plant phenotyping is a scientific process of identifying physical plant characteristics and function (known as the phenotype) that can be jointly affected by genotype and environment. In recent years, computer vision technologies with deep learning have been widely used in plant phenology and phenotyping to improve plant productivity [36–38].

(f) Species Recognition
CVML can be used for faster detection and classification of plant species to reduce the classification time without human effort. A research study based on the identification and classification of three legume species (white beans, red beans, and soybean) through leaf vein patterns was presented in [39].

(g) Yield Prediction
Yield prediction has become one of the most popular research topics in precision agriculture as it has outperformed the simple prediction based traditional

methods for crop yield. Several surveys on yield prediction using machine learning algorithms have been conducted during the past few years [40–42].

(h) Water Management
Water management has significant impacts on hydrological, climatological, and agronomical balance in agriculture. Several machine learning algorithms have been developed to build an effective regular irrigation system based on weather conditions and evaporation [43–45].

(i) Soil Management

Machine learning algorithms are used to study evaporation processes and measure soil moisture and temperature for a better understanding of the essential eco-elements in agriculture [46, 47].

3 Challenges and Future Scopes

As an emerging technology, computer vision combined with machine learning algorithms has become a crucial factor in the development of agricultural efficiency with spacious prospects in agricultural research. With extensive progress in machine vision technology, the complexity of agricultural automation has been minimized to a great extent. In recent years, computer vision-based machine learning algorithms assembled with GPUs have been widely applied in various agricultural disciplines, as GPU is mainly used to increase the computing power while processing high-density data [7]. However, computer vision technology requires more skilled professionals.

Besides, heterogeneous elements in different environments have a diverse impact on crop production and harvesting. In that case, computer vision technology has been rigorously applied in complicated environmental situations for its robustness and high complexity. But at the same time, most of the existing vision-based methods are only implemented in a laboratory or a customized environment for experimentation that resulting in a huge inconsistency between the experimental and actual data.

Existing machine vision techniques are limited to completely control specific crop diseases and pest attacks. Hence, generalized methods need to be developed.

Moreover, the scarcity of large-scale universal datasets has diminished the research progress in this sector. Thus, agricultural databases should be extended for research to overcome image acquisition problems in different environmental conditions. Besides, machine vision will require further renovation in terms of unmanned technology, as it is still not applicable to solve all issues in automated farm management and precision agriculture. Thus, scopes for machine vision technology will be expanded in the future to be more versatile by establishing large-scale datasets.

4 Conclusion

This chapter reported an introduction to computer vision and machine learning applications in agriculture that can be served as a strong reference by illustrating the latest advancements in agricultural machine vision applications. It will certainly motivate researchers in contributing to the development of agricultural tools for crop health growth monitoring, disease and pest detection and control, weeding, irrigation, crop management, and harvesting with low cost and high efficiency. Finally, it can be concluded that in the future, machine vision technology associated with large-scale datasets will be immensely used in every aspect of agricultural automation to overcome the current challenges in agriculture.

References

1. Liakos, K.G., Busato, P., Moshou, D., Pearson, S., Bochtis, D.: Machine learning in agriculture: A review. Sensors **18**(8), 1–29 (2018)
2. Bochtis, D.D., Sørensen, C.G.C., Busato, P.: Advances in agricultural machinery management: A review. Biosys. Eng. **126**, 69–81 (2014)
3. Hunter, M., Smith, R., Schipanski, M., Atwood, L., Mortensen, D.: Agriculture in 2050: Recalibrating targets for sustainable intensification. Bioscience **67**(4), 386–391 (2017)
4. Wu, X., Guo, J., Han, M., Chen, G.: An overview of arable land use for the world economy: From source to sink via the global supply chain. Land Use Policy **76**, 201–214 (2018)
5. Aubert, B.A., Schroeder, A., Grimaudo, J.: IT as enabler of sustainable farming: An empirical analysis of farmers' adoption decision of precision agriculture technology. Decis. Support Syst. **54**(1), 510–520 (2012)
6. Gomes, J.F.S., Leta, F.R.: Applications of computer vision techniques in the agriculture and food industry: A review. Eur. Food Res. Technol. **235**(6), 989–1000 (2012)
7. Rehman, T.U., Mahmud, M.S., Chang, Y.K., Jin, J., Shin, J.: Current and future applications of statistical machine learning algorithms for agricultural machine vision systems. Comput. Electron. Agric. **156**, 585–605 (2019)
8. Mahajan, S., Das, A., Sardana, H.K.: Image acquisition techniques for assessment of legume quality. Trends Food Sci. Technol. **42**(2), 116–133 (2015)
9. Vithu, P., Moses, J.A.: Machine vision system for food grain quality evaluation: A review. Trends Food Sci. Technol. **56**, 13–20 (2016)
10. Kirk, D.B., Hwu, W.W.: "Programming Massively Parallel Processors," A Hands-on Approach, 3rd edn. Morgan Kaufmann, San Francisco, CA, USA (2016)
11. Tripicchio, P., Satler, M., Dabisias, G., Ruffaldi, E., Avizzano, C.A.: Towards smart farming and sustainable agriculture with drones. Int. Conf. Intell. Environ. 140–143 (IEEE, 2015)
12. Wjtowicz, M., Wjtowicz, A., Piekarczyk, J.: Application of remote sensing methods in agriculture. Commun. Biometry Crop Sci. **11**(1), 31–50 (2016)
13. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. Adv. Neural. Inf. Process. Syst. **25**(2), 1097–1105 (2012)
14. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich: Going deeper with convolutions. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–9, Boston, MA (2015).
15. Simonyan, K., Zisserman, A.: Very Deep Convolutional Networks for Large-Scale Image Recognition (2014). Available Online: https://arxiv.org/pdf/1409.1556.
16. Cheng, G., Han, J., Lu, X.: Remote sensing image scene classification: Benchmark and state of the art. Proc. IEEE **105**(10), 1865–1883 (2017)

17. Mavridou, E., Vrochidou, E., Papakostas, G.A., Pachidis, T., Kaburlasos, V.G.: Machine vision systems in precision agriculture for crop farming. J. Imaging **5**(12), 1–32 (2019)
18. Li, K., Lian, H., Deun, R.V., Brik, M.G.: A far-red-emitting $NaMgLaTeO_6$:Mn^{4+} phosphor with perovskite structure for indoor plant growth. Dyes Pigm. **162**, 214–221 (2019)
19. Shimizu, H., Heins, R.: Computer-vision-based system for plant growth analysis. Trans. ASAE **38**(3), 959–964 (1995)
20. Tombe, R.: Computer vision for smart farming and sustainable agriculture. In: 2020 IST-Africa Conference (IST-Africa), Kampala, Uganda, pp. 1–8 (2020)
21. Maharlooei, M., Sivarajan, S., Bajwa, S.G., Harmon, J.P., Nowatzki, J.: Detection of soybean aphids in a greenhouse using an image processing technique. Comput. Electron. Agric. **132**, 63–70 (2017)
22. Xiaolong, L., Ma, Z., Bienvenido, F., Feng, Q., Haiguang, W., Álvarez-Bermejo, J.A.: Development of automatic counting system for Urediospores of wheat stripe rust based on image processing. Int. J. Agric. Biol. Eng. **10**(5), 134–143 (2017)
23. Wang, G., Sun, Y., Wang, J.: Automatic image-based plant disease severity estimation using deep learning. Comput. Intell. Neurosci. **2017**, 1–8 (2017)
24. Liu, H., Chahl, J.S.: A multispectral machine vision system for invertebrate detection on green leaves. Comput. Electron. Agric. **150**, 279–288 (2018)
25. Zhong, Y., Gao, J., Lei, Q., Zhou, Y.: A vision-based counting and recognition system for flying insects in intelligent agriculture. Sensors **18**(5), 1–19 (2018)
26. Tellaeche, A., Pajares, G., Burgos-Artizzu, X.P., Ribeiro, A.: A computer vision approach for weeds identification through support vector machines. Appl. Soft Comput. **11**(1), 908–915 (2011)
27. Sabzi, S., Abbaspour-Gilandeh, Y., Garcia-Mateos, G.: A fast and accurate expert system for weed identification in potato crops using metaheuristic algorithms. Comput. Industry **98**, 80–89 (2018)
28. Pantazi, X.E., Tamouridou, A.A., Alexandridis, T.K., Lagopodi, A.L., Kashefi, J., Moshou, D.: Evaluation of hierarchical self-organising maps for weed mapping using UAS multispectral imagery. Comput. Electron. Agric. **139**, 224–230 (2017)
29. Muppala, C., Guruviah, V.: Machine vision detection of pests, diseases and weeds: A review. J. Phytol. **12**, 9–19 (2020)
30. Yuan, T., Xu, C.-G., Ren, Y.-X., Feng, Q.-C., Tan, Y.-Z., Li, W.: Detecting the information of cucumber in greenhouse for picking based on NIR image. Guang Pu Xue Yu Guang Pu Fen Xi **29**(8), 2054–2058 (2009)
31. Davidson, J.R., Silwal, A., Hohimer, C.J., Karkee, M., Mo, C, Zhang, Q.: Proof-of-concept of a robotic apple harvester. In: International Conference on Intelligent Robots and Systems (IROS). IEEE/RSJ, Daejeon, pp. 634–639 (2016)
32. Zhang, Q., Chen, S., Yu, T., Wang, Y.: Cherry recognition in natural environment based on the vision of picking robot. IOP Conf. Series Earth Environ. Sci. **61**(1), 1–6 (2017)
33. Patel, K.K., Kar, A., Jha, S.N., Khan, M.A.: Machine vision system: A tool for quality inspection of food and agricultural products. J. Food Sci. Technol. **49**(2), 123–141 (2012)
34. Saldaña, E., Siche, R., Luján, M., Quevedo, R.: Review: Computer vision applied to the inspection and quality control of fruits and vegetables. Braz. J. Food Technol. **16**(4), 254–272 (2013)
35. Bhargava, A., Bansal, A.: Fruits and vegetables quality evaluation using computer vision: A review. J. King Saud Univ. Computer Inf. Sci. 1–15 (2018)
36. Mochida, K., Koda, S., Inoue, K., Hirayama, T., Tanaka, S., Nishii, R., Melgani, F.: Computer vision-based phenotyping for improvement of plant productivity: A machine learning perspective. GigaScience **8**(1), 1–12 (2018)
37. Li, Z., Guo, R., Li, M., Chen, Y., Li, G.: A review of computer vision technologies for plant phenotyping. Comput. Electron. Agric. **176**, 1–21 (2020)
38. Chandra, A.L., Desai, S.V., Guo, W., Balasubramanian, V.N.: Computer vision with deep learning for plant phenotyping in agriculture: A survey. Adv. Comput. Commun. 1–27 (2020)

39. Grinblat, G.L., Uzal, L.C., Larese, M.G., Granitto, P.M.: Deep learning for plant identification using vein morphological patterns. Comput. Electron. Agric. **127**, 418–424 (2016)
40. Chlingaryan, A., Sukkarieh, S., Whelan, B.: Machine learning approaches for crop yield prediction and nitrogen status estimation in precision agriculture: A review. Comput. Electron. Agric. **151**, 61–69 (2018)
41. Elavarasan, D., Vincent, D.R., Sharma, V., Zomaya, A.Y., Srinivasan, K.: Forecasting yield by integrating agrarian factors and machine learning models: A survey. Comput. Electron. Agric. **155**, 257–282 (2018)
42. van Klompenburg, T., Kassahun, A., Catal, C.: Crop yield prediction using machine learning: A systematic literature review. Comput. Electron. Agric. **177**, 1–18 (2020)
43. Mohammadi, K., Shamshirband, S., Motamedi, S. Petkovic, D. Hashim, R., Gocic, M.: Extreme learning machine based prediction of daily dew point temperature. Comput. Electron. Agric. **117**, 214–225 (2015)
44. Patil, A.P., Deka, P.C.: An extreme learning machine approach for modeling evapotranspiration using extrinsic inputs. Comput. Electron. Agric. **121**, 385–392 (2016)
45. Mehdizadeh, S., Behmanesh, J., Khalili, K.: Using MARS, SVM, GEP and empirical equations for estimation of monthly mean reference evapotranspiration. Comput. Electron. Agric. **139**, 103–114 (2017)
46. Morellos, A., Pantazi, X.-E., Moshou, D., Alexandridis, T., Whetton, R., Tziotzios, G., Wiebensohn, J., Bill, R., Mouazen, A.M.: Machine learning based prediction of soil total nitrogen, organic carbon and moisture content by using VIS-NIR spectroscopy. Biosys. Eng. **152**, 104–116 (2016)
47. Nahvi, B., Habibi, J., Mohammadi, K., Shamshirband, S., Razgan, O.S.A.: Using self-adaptive evolutionary algorithm to improve the performance of an extreme learning machine for estimating soil temperature. Comput. Electron. Agric. **124**, 150–160 (2016)

Robots and Drones in Agriculture—A Survey

Rabeya Basri, Fahmida Islam, Sumaita Binte Shorif, and Mohammad Shorif Uddin

1 Introduction

Agriculture is a crucial factor that has a significant contribution to the global economy, as more than 60% of our population is entirely dependent on agriculture for survival [1]. Additionally, continuous expansion of urbanization, which is responsible for the gradual destruction of the land area for cultivation, causes large-scale damage to agriculture [2]. Despite being the leading source of food and income, agronomy is a tremendously time-consuming, labor-intensive, and slow speed process. Thus, agricultural robotics has been introduced to eliminate these barriers and increase the accuracy of an efficient autonomous agricultural system [3]. During the past few decades, robotics has immensely been applied in different fields, including smart home, medical research and diagnosis, manufacturing, agricultural industry [4–7], and so on. An agricultural robot is such an automated machine, which operates different computational algorithms to increase production efficiency by considering the agro-products as objects, based on environmental perceptions [8]. In recent years, precision agriculture has emerged with artificial technologies for the automation of farming processes to minimize labor-intensive work and time [9]. This precision concept has brought a tremendous change in the design of agricultural tools by

R. Basri · F. Islam · S. B. Shorif · M. S. Uddin (✉)
Department of Computer Science and Engineering, Jahangirnagar University, Savar, Dhaka, Bangladesh
e-mail: shorifuddin@gmail.com

R. Basri
e-mail: rabeyabasri3@gmail.com

F. Islam
e-mail: jharnaju@yahoo.com

S. B. Shorif
e-mail: sumaita.bs@gmail.com

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_2

connecting them with small smart devices such as different types of sensors, drones, trackers, etc. which can easily detect, spray, weed, and pick crops. An agricultural robot can be designed by using software and GSM to interface the robot with a computer [10]. Agricultural production has been increased to a great extent by the substantial usage of these agricultural robots. In this paper, we review the progress of research on agricultural robots in terms of design, classification, grafting, picking, weeding, spraying, harvesting [11–15], etc. along with their features and operations. Furthermore, we instantiate a summary of robotic development with technical challenges and future scopes in agricultural robotics.

The remainder of this chapter is structured as follows: Sect. 2 presents a basic architecture and classification of agricultural robots with their functionalities. Sections 3 and 4 illustrate various applications of agricultural robots and drones, respectively. Section 5 shows the commercialization and current challenges of agricultural robots and drones. Finally, Sect. 6 provides a conclusion.

2 Robotics Basic

Robotics in agriculture had been introduced to replace traditional manual farming methods for more production with improved quality using less effort. A brief elucidation of such robotic mechanisms is as follows.

2.1 Robotic Mechanism

Agricultural robots are obliged to be designed by using an efficacious computational model with excellent algorithms and smart gadgets, as they have to operate on rough terrain and overcome several technical challenges. Researchers primarily focused on the following requirements for agricultural robotics mechanism:

- **Specific Path Navigation**: Reliable path navigation is indispensable in robotic design for proper plantation. Fuzzy logic [16–18] was widely used in path tracking and navigation.
- **Image Processing**: In a vision-based system, specifically a camera is used for image acquisition, and then a segmentation process is used to find the path through the plants for the navigation of robots to do specific tasks based on image-based recognition [19].
- **Obstacle Avoidance**: In a robust obstacle avoidance system, GPS, ultrasonic, infrared, and vision sensors are used to handle the navigation on rough terrain by avoiding wheel slip [20–23].
- **Mechanical Design**: Microcontrollers with various types of sensors are used as hardware in operating drives and control the robot for multitasking [24].

The required hardware, sensors, and software requirements to design an agricultural robot for an efficient agricultural system [25–27] shown in Table 1 and Fig. 1.

Table 1 System requirements of an agricultural robot

Module	Functional role	Internal system	Specification	
Hardware	Transducers and actuators	Mechanical system	Pinion belt	
			Steeper motor	
		Electrical and control system	Motor driver	
			Arduino board	
			Relays	
			Battery	
			Voltage divider	
		Other tools system	Mount (Solid dispensing mechanism)	
			Vacuum pump (Liquid dispensing mechanism)	
			Seeder	
			Driller	
			Weeder	
			Fertilizer	
System and sensors	Processing in arduino		Arduino IDE	
			Processor (e.g., Raspberry Pi)	
			USB camera	
			GPS	
			WiFi	
			Sensors	Ultrasonic sensors
				Orientation sensor
				Soil moisture sensor
				Temperature sensor
				Rainfall sensor
				Wind sensor

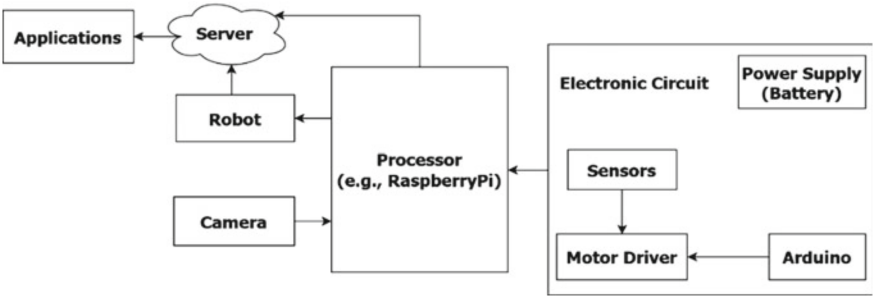


Fig. 1 System block diagram of an agricultural robot

2.2 Agricultural Robot Classification

Due to the vast application of modern technologies, researchers have recently shown a high interest in the research and development of agricultural robots. According to the use of robots in diverse agricultural tasks, agricultural robots can be categorized mainly as outdoor and indoor robots, which can be further classified according to their operations. Table 2 illustrates the classification of agricultural robots along with their specific functions [28, 29].

3 Robots in Agricultural Applications

During the past few decades, even though there was a presence of manpower, robots have been used to perform several challenging agricultural tasks such as autonomous path navigation, grafting, seeding, weeding, spraying, harvesting, and so on. All these tasks should be studied well for better understanding.

3.1 Robots in Path Navigation

Automatic path navigation is the most critical task when it comes to high-value crops. It requires localization, tracking, mapping, motion control, and path planning. Robots generally navigate the path by using cameras to take pictures of the path and plan the desired path by recognizing the color of the plants, usually detecting green or not. Path navigation and planning in agriculture have been studied well, and research in this sector is developing day by day. A brief review of different path navigation modules is described as follows and a basic navigation strategy is shown in Fig. 2.

Table 2 Classification of agricultural robots

Category	Class	Name	Function
Outdoor robot	Field robots	Autonomous navigator	Automatically runs on the terrain to find a suitable path for farming using a camera. For instance, a tractor with GPS automatically avoids any obstacles during plantation
		Disease detection and spray robots	Moves along the crop rows in the field to detect the crop diseases or identify any pests and sprays pesticides to control the disease at an early stage
		Weeding robot	Detects weeds and unwanted plants and uses herbicides to kill them without causing any harm to the desired crops
		Drone	Is an unmanned aerial vehicle that visually monitors the health of the crop using a camera and generally used to spray pesticides, herbicides, fertilizers
	Fruit and vegetable harvesting robots	Grafting robot	Used for fruit and vegetable grafting
		Picking robot	Identifies and picks the ripe fruits and vegetables using sensors
		Sorting robot	Classifies fruits and vegetables according to their size, shape, color, and other attributes
	Forest robots [30]	Forest robot	Moves through the forest using a pacing mechanism to create a 3D map for easily counting the number of trees, classifying them, measuring their diameter, and examining the pathology of the trees
	Animal husbandry robots	Milk robot	Used to select cows, which are suitable for milking by identifying their nipple position

(continued)

Table 2 (continued)

Category	Class	Name	Function
		Lumbering robot	Cleans up different slopes and terrain using a hydraulic system
Indoor robot	Harvesting robots	Greenhouse harvesting robot	Uses vision-based machines for movement through greenhouse aisles to harvest some sensitive crops like tomatoes and strawberries
	Material handling robots	Greenhouse Material Handler	Performs plant spacing and optimizes plant placement to reduce production costs by controlling the use of water, pesticides, herbicides, and fertilizers for the high-quality plant

(a) **GPS-Based Navigation**

Hellström [31] combined GPS with Inertial Navigation System (INS) and other sensors to collect accurate information for navigation by measuring the derivatives of robot position using gyros and accelerometers [31]. Nagasaka et al. [32] presented an automated rice transplanter that accurately transplanted rice using a real-time kinematic global positioning system (RTKGPS) and fiber optic gyroscope (FOG) sensor to measure the direction, and a simple steering controller in the desired path less than 12 cm. Hellström [33] introduced a superior algorithm for path tracking, where a radar performed obstacle avoidance and a real-time kinematic differential GPS/GLONASS completed localization. Li et al. [34] developed a conceptual framework along with reviewing the guidance of an autonomous agricultural vehicle consisting of navigation sensors with GPS, computational methods, and steering controllers. Ringdahl et al. [35] designed an automated vehicle to track two different paths more accurately by using a forwarder with GPS and a gyro. However, this system can track only previously demonstrated paths.

(b) **Vision-Based Navigation**

Zhao et al. [19] presented a robot that used Hough transform to track the path from the continuous sample images captured by the camera. Gottschalk et al. [36] developed a machine to navigate between two-crop rows for automatic field inspection through image segmentation, classification, and geometrical line extraction by using a webcam. Wang et al. [37] presented an improved path navigation framework against local map navigation based on near to far perception [37]. Besides, a large quantity of meticulous research has been conducted to develop an automated vision-based navigation system [38–41]. Weiss et al. [42] introduced a stereo vision system to map 3D filed images for autonomous navigation. Ball et al. [43] defined a cost-effective vision-based navigation and obstacle detection system to spread a sustainable domain of agriculture [43].

(c) **Computational Method-Based Navigation**

Several computational algorithms, such as Kalman filter, fuzzy logic, neural network, genetic algorithm, etc. are used to extract key information for autonomous navigation.

(i) *Kalman filter*
It is based on the strong mathematical formula for data fusion of multiple sensors in real time. Previously, GPS combined with other sensors using Kalman filter was used to improve the accuracy of an estimated position [44, 45]. Kalman filter was performed using laser data with the data from an encoder for mobile robot localization in the orchard environment [46]. Laser scanner combined with other sensors like odometer and inertial measurement unit (IMU) improved the location estimation using the Kalman filter [47]. An autonomous row guidance simulator was presented for a differentially wheeled robot by estimating the position of the row using a Kalman filter [48].
(ii) *Fuzzy logic*
It is used to control a robot by making inferences under complex and nonlinear conditions. Boubertakh [49] designed a fuzzy controller for path navigation and obstacle avoidance to seek the destination without being stuck. Yousfi [50] applied the gradient method to optimize a Takagi–Sugeno fuzzy controller for navigation and obstacle avoidance. It is vital that the robot can recognize and avoid the obstacles for the successful implementation of a navigation system. With this view, a fuzzy logic controller was designed for inter-row navigation using one-dimensional (1D) ranging sensors in [51].
(iii) *Neural network and genetic algorithm*
A hybrid neural network architecture was presented for robot navigation in [52]. A genetic algorithm-based system was presented for optimal path planning with an obstacle avoidance mechanism [53]. Nicholas et al. [54] presented a biologically inspired spiking neural network in optimal pathfinding for robots. Motlagh et al. [55] proposed a smart robot for obstacle avoidance to seek the desired goal using neural network mechanisms [55].

3.2 Robots in Crop Production

(a) **Grafting Robot**

Japan first introduced grafting robots while South Korea and China were also the pioneers in grafting robot research. In 1986, Japan developed automatic vegetable grafting robots for research and development. Among them, the first commercial model was launched in the market in 1993 with a success rate of 90% [56–58]. Being inspired by the successful efficiency rate, several companies in Asia, Europe, and the

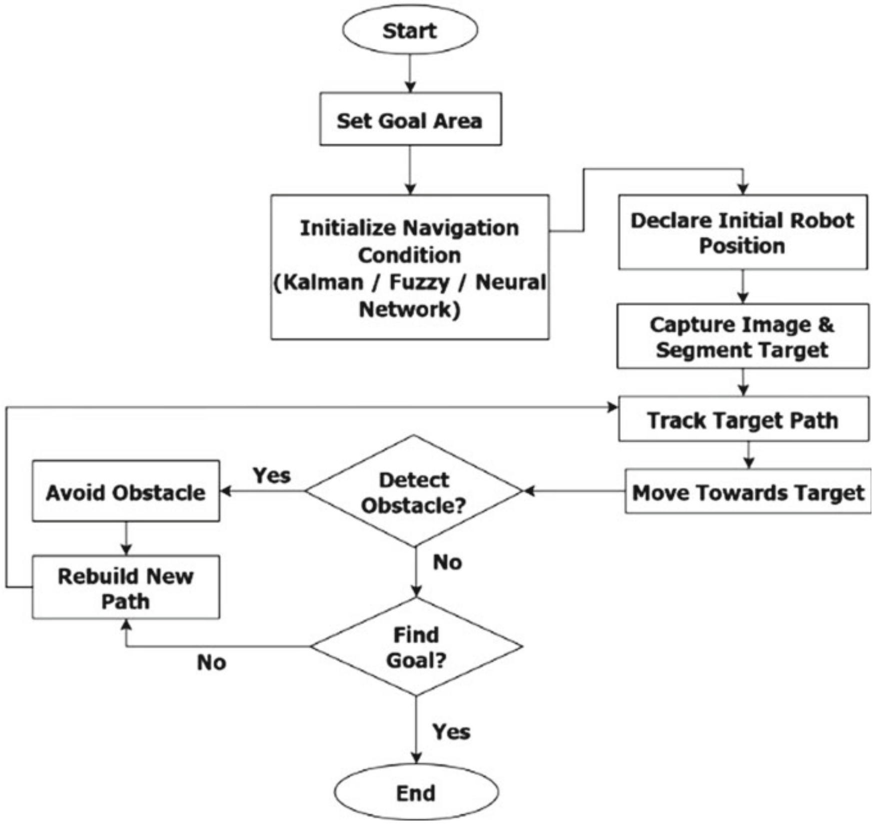


Fig. 2 Schematic flowchart of a robot navigation strategy

USA started to produce different types of commercial robot models of automatic fruit and vegetable grafting [59–64]. Li et al. [65] developed a plant let-cutting mechanism for the grafting robot with an efficiency of 99.14%. On the other hand, Zhao et al. [66] applied a scion cutting mechanism for a sapling grafting robot and obtained 96.5% efficiency. Jiang [67] developed a grape sapling robot for a stick-to-stick grafting. A grafting robot was presented using the cutting grafting method for camellia oleifera seedlings in [68].

(b) **Picking Robot**

California Machinery Company introduced an automatic tomato picking robot in 2004. This automatic tomato picking robot was used to pick fruits and separate them from leaves into a sorting bin to further process the picked leaves as fertilizer. Nezhad et al. [69] introduced a tomato picking robot that can identify and pick tomatoes through image processing. Feng et al. [70] developed an intelligent tomato picking robot to reduce labor effort harvesting fresh tomatoes with a success rate of 83.9%. Xiong et al. [71] presented a novel mechanism for faster picking of strawberries using a cable-driven gripper. An automated cucumber picking robot was presented in [12]. Ashwini [72] designed a strawberry picking robot capable of picking a large number of mature fruits at a minimum time with less effort.

3.3 Robots in Weed Removal and Disease and Pest Control

(a) **Weeding Robot**

Weeds or unwanted plants must be controlled for faster growth of crops using automatic weeding [73]. A four-wheeled weeding robot was developed that was composed of a diesel engine, hydraulic transmission, and was able to steer 360° to detect and remove weeds automatically [74]. A co-robot system along with odometry was described for inter-row weeding without damaging the crop rows [75]. Fennimore et al. [76] focused on the improvement of weeding efficiency with a reduction in production cost. Bechar and Vigneault [77] discussed the development of weeding robots [77] for real-field operations. Kunz et al. [78] introduced integrated weed management in sugar beet, maize, and soybean fields, combining multiple tactics to increase weeding performance [78]. Chang and Lin [79] proposed a vision-based agricultural robot for weeding in real-time with average plants and weeds classification rate of 90% or higher. It can also perform smart watering while maintaining the moisture content of the deep soil at an efficiency of 80%. Steward et al. [80] emphasized the development of efficient weed robots including perception system and weed control mechanisms. Wu et al. [81] used a multi-camera system for removing weeds by classifying plants and weeds.

(b) **Disease and Pest Detector**

Detecting diseases and controlling pests at an early stage is inevitable for high-quality crop production. Previously farmers used different strategies to detect crop diseases [82]. Hengstler et al. [83] designed a smart visual surveillance system (Mesh Eye) to detect diseases of fruits and identify the affected area by determining the shape and color of the leaf and stem [83]. Additionally, one major problem was controlling pest nuisance, caused by harmful insects or germs that destroyed crops. Pests cannot entirely be eliminated but can be effectively controlled to reduce their nuisance. Thus, early detection of pests and monitor them are essential to control them for the prevention of crop damages. Laothawornkitkul et al. [84] used a potential electronic-nose technology for remote sensing and monitoring of diseases and pests [84]. Pest identification on leaves along with automatic spraying was performed by using a vision-based system [85]. Camargoa et al. [86] proposed a visual method, capable of detecting all visual symptoms of plant diseases by analyzing colored images [86]. López et al. [87] presented an effective pest control technique, known as insect traps to efficiently perform automatic pest monitoring and inspection using high-resolution images [87]. Francis et al. [88] used soft computing techniques to identify leaf diseases in pepper plants. Yazgac et al. [89] applied a signal processing method to detect the sunn pests in wheat and barley by sound, but this method was useful only for a single leaf [89]. Gonzalez-de-Santos et al. [90] presented a fleet of heterogeneous ground and aerial robots that were used for effective weed and pest control to reduce the use of chemical substances.

(c) **Pest Control and Spraying Robot**

In recent years, different types of pesticides and chemical sprays have been used to control pests, which are very dangerous for humans resulting in skin cancer, asthma, or other chronic diseases. In that case, an automated robotic system can be used to spray pesticides to avoid human contact and save time, as robots are programmed to spray pesticides on crops only if they can detect pests. Blackmore et al. [91] applied a multi-purpose robotic sprayer that can operate automatically according to the weather conditions. Jian-sheng [92] proposed a wireless controlled robot for spraying pesticides. Pilli et al. [93] designed a robot that was capable of moving along the intercrop rows to detect diseases and spraying pesticides automatically and showed promising results in cotton and groundnut fields. Sharma and Borse [94] described a mobile robot to monitor plant growth and detect disease with the spraying mechanism for pesticides, fertilizers, and water. Sudha et al. [95] proposed an automated technique to detect pests and spray pesticides based on a pre-defined threshold value to indicate the pesticide level. Chaitanya et al. [96] designed an autonomous robot, capable of spraying pesticides in a limited quantity only when pests were detected.

3.4 Robots in Crop Harvesting

For many years, robots have been continuously used for harvesting a variety of fruits and vegetables such as apple, grapes, watermelon, tomatoes, cucumbers, etc. Ceres et al. [97] proposed a new robot (Agribot) for fruit harvesting in a particularly unstructured environment. Yanbin et al. [98] demonstrated the current development of automatic fruit harvesting robots in horticulture and experimented on apple, kiwi, tomato, and sweet pepper. Onishi et al. [99] presented an automated fruit detection and harvesting robot using a single shot multi-box detector and a stereo camera and found a performance of more than 90%. Some robots are described below based on their application for specific fruits.

(a) **Apple Harvesting**

Yuan et al. [100] applied an ant colony algorithm to improve the performance of the apple harvesting system and found an optimal result. Lv et al. [101] proposed a vision- based apple harvesting robot to efficiently recognize apples using video. De-An et al. [102] presented a 5 DOF (*degrees of freedom)* robot to detect, locate, and pick apples automatically using the support vector machine algorithm.

(b) **Tomato Harvesting**

Li et al. [103] presented a tomato harvesting robot using the joint analysis of Kinematics. Wang et al. [105] designed a vision based 4-DOF tomato harvesting robot [104]. Liu et al. [105] developed a coordinated hand-arm for the tomato harvesting system with a success rate of over 70%. Yaguchi et al. [106] developed a tomato harvesting robot with a stereo camera to measure the depth in the short range by utilizing the infinity rotational joint. A new picking end-effector was developed for automatic tomato harvesting in [107] representing three modules: tomato holding, stem holding, and tomato picking.

(c) **Watermelon Harvesting**

Heavy fruits like watermelon harvesting require a smart and quite strong robotic system of carrying a heavy load. A high-speed control watermelon harvesting system was implemented with great efficiency in [108].

(d) **Strawberry Harvesting**

Harvesting of small and sensitive fruits, e.g., strawberries through robots, was studied in [8, 109, 110]. A grape harvesting robotic system was designed by using different end effectors to perform diverse tasks such as harvesting, thinning, spraying, and bagging in [8]. On the other hand, a multifunctional strawberry harvesting robot was also developed for harvesting, spraying, and grading in [109]. In addition, a robot was designed based on machine vision and sonar technology for table-top culture with minimum production cost in [110].

Besides the above harvesting robots, a 3D visual asparagus harvesting robot [111] was developed for better recognition, where its arm grasped the asparagus to cut, and a vision-based robot was designed to localize and classify lettuce for harvesting using state-of-the-art approaches in [112]. Some extensive details on the development, practicability, and applications of agricultural robots are available in references [113–117]. Different types of agricultural robots are shown in Fig. 3.

4 Drones in Agriculture

As a blessing of modern technology, drones (unmanned aerial vehicles, UAVs) [118–132] have been used to perform various agricultural tasks such as crop health monitoring, weed identification, pesticide spraying, forestry, and wildlife monitoring as well as in precision agriculture. Several agricultural drones are illustrated in Fig. 4.

Kurkute et al. [130] listed more than 250 models of UAVs to help in selecting the perfect one for the agricultural task. Table 3 represents the classification of drones with their basic features and Table 4 summarizes the applications of agricultural drones.

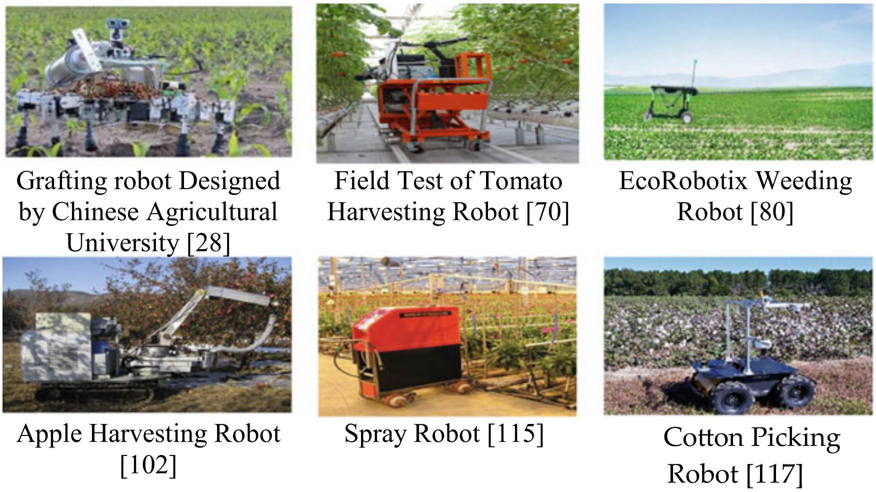


Fig. 3 Examples of different types of agricultural robots

Fig. 4 Example of agricultural drones

Table 3 Classification of agricultural drones

Type	Feature and suitability
Rotary drones	Classified as a single rotor and multi-rotor drone according to the number of rotors. Useful for smaller field scouting and can fly up to 20 min during normal wind pressure
Fixed wing drones	Fly at high speed with longer battery lives like more than 20 min or an hour. Good for irrigation and plant growth analysis
VTOL (Vertical take-off and landing) drones	It can perform take-off and land vertically. Suitable for spraying crops in a small range of areas with the shorter flight time

Table 4 An overview of agricultural drone applications

Application		References	Research output
Crop analysis	Monitoring	Zheng et al. [133]	It developed a mechanism to get rice nitrogen status from unmanned aerial vehicles (UAVs) for crop monitoring
		Reinecke et al. [134]	It developed a technique utilizing UAVs by using specific cameras for early detection of pests and water shortages to improve crop health and harvest size to maximize the production
	Crop protection	Psirofonia et al. [135]	It developed a new application for UAVs in agriculture for the protection of olive trees by mapping of trees and identification of visible signs of infestation by pest or disease, and also counting the insects by making cooperation between the UAV with electronic traps
Spraying	Pesticide	Yallappa et al. [136]	It designed and developed a drone-mounted pesticide sprayer that is useful on crops such as rice fields, orchard crops, and crops in terrain lands
	Fertilizer	Pharne et al. [137]	It developed a cost-effective system that sprayed pesticide and fertilizer on the crop using satellite data with a test accuracy of 87% by using a minimum amount of time
Remote sensing		Xiang et al. [138]	It developed an agricultural remote sensing system based on a UAV. It did experimentation on a turfgrass field to monitor the spatial and temporal changes by capturing multi-spectral images in that particular area and found good performance
		Senthilnath et al. [139]	It developed a tomato detection mechanism using RGB images from UAVs using spectral-spatial classification

5 Commercialization and Current Challenges of Agricultural Robots

Several commercial agricultural robots [140–142], featured with different localization techniques like GPS, vision, laser, sensor-based navigation, etc., were developed in Germany, Denmark, Finland, USA, Australia, New Zealand, India, and many other countries for commercial use. Despite having a remarkable advancement in agricultural robotics, it is still dealing with some challenges. Farmers are still handling several technical challenges while using autonomous agricultural vehicles in terms of automated decision making and prediction for digital farming. This industry is still underserved to have an advanced vision-based system to enable ultra-level precision agriculture. We need more research initiatives and robust infrastructures with a reliable wireless connection, use IoT, human–robot interaction tools to explore numerous future scopes of smart robots in agriculture. Another significant issue is that due to the excessive cost of robots along with spare expenses, these are not easily affordable by farmers. Thus, an open-source framework will highly be required to make these robots more affordable to fill up the agricultural demand. Though drones have already been applied for crop monitoring, early detection of weed and pest, and spraying pesticides or herbicides, however, more work is required to make it robust and precise.

6 Conclusion

Automation in agriculture using intelligent robotic systems is one of the most challenging tasks that has already grabbed the attention of researchers due to its diverse practical applications for commercial use. This chapter is reported on an extensive review of the time-to-time development of robotic applications including their classification and some useful practical applications in agriculture. Agriculture is the most valuable source of income, as more than 60% of the world population earn their livelihood from farming. Thus, technical scopes of agricultural robotics should be spread in cities, where farming scopes are limited. More automation in agricultural robots is expected to help the farmers by efficiently increasing crop productivity using solar energy that can work for many hours without any break. Moreover, the invention of drones or aerial devices has already brought a huge revolution in digital farming, which will extend scopes for smart crop management such as, crop scouting, crop monitoring, weed, and pest control, spraying, and selective harvesting in precision agriculture.

References

1. Alston, J.M., Pardey, P.G.: Agriculture in the Global Economy. J. Econ. Perspect. **28**(1), 121–146 (2014)
2. Calabi-Floody, M., Medina, J., Rumpel, C., Condron, L.M., Hernandez, M., Dumont, M., Mora, M.L.: Smart fertilizers as a strategy for sustainable agriculture. Adv. Agron. **147**, 119–157 (2018)
3. Ahmed, H., Juraimi, A.S., Hamdani, S.M.: Introduction to robotics agriculture in pest control: A review. Pertanika J. Sch. Res. Rev. **2**(2), 80–93 (2016)
4. Lee, K.W., Kim, H.R., Yoon, W.C., Yoon, Y.S., Kwon D.S.: Designing a human-robot interaction framework for home service robot. In: International Workshop on Robots and Human Interactive Communication, pp. 286–293. IEEE (2005)
5. Barbash, G.I., Glied, S.A.: New technology and health care costs—The case of robot-assisted surgery. N. Engl. J. Med. **363**(8), 701–704 (2010)
6. Wang, W., Li, R., Chen, Y., Diekel, Z.M., Jia, Y.: Facilitating human-robot collaborative tasks by teaching-learning-collaboration from human demonstrations. In: IEEE Transactions on Automation Science and Engineering, pp. 1–14. IEEE (2018)
7. Bechar, A., Edan, Y.: Human-robot collaboration for improved target recognition of agricultural robots. Ind. Robot **30**(5), 432–436 (2003)
8. Monta, M., Kondo, N., Shibano, Y.: Agricultural robot in grape production system. In: International Conference on Robotics and Automation, vol. 3, pp. 2504–2509. IEEE, Nagoya, Japan (1995)
9. Amer, G., Mudassir, S.M.M., Malik, M.A.: Design and operation of Wi-Fi agribot integrated system. In: International Conference on Industrial Instrumentation and Control (ICIC), pp. 207–212. IEEE, Pune (2015)
10. Auat Cheein, F.A., Carelli, R.: Agricultural robotics: Unmanned robotic service units in agricultural tasks. Ind. Electron. Mag. **7**(3), 48–58, IEEE (2013)
11. Kurata, K.: Cultivation of grafted vegetables II. Development of grafting robots in Japan. HortScience **29**, 240–244 (1994)
12. Van Henten, E.J., Van Tuijl, B.A.J., Hemming, J., Kornet, J.G., Bontsema, J., Os, E.A.V.: Field test of an autonomous cucumber picking robot. Biosys. Eng. **86**(3), 305–313 (2003)
13. Bawden, O., Kulk, J., Russell, R., McCool, C., English, A., Dayoub, F., Lehnert, C., Perez, T.: Robot for weed species plant-specific management. J. Field Robot. **34**(6), 1179–1199 (2017)
14. Sammons, P.J., Furukawa, T., Bulgin, A.: Autonomous pesticide spraying robot for use in a greenhouse. Aust. Conf. Robotics Autom. 1–9 (2005)
15. Tanigaki, K., Fujiura, T., Akase, A., Imagawa, J.: Cherry-harvesting robot. Comput. Electron. Agric. **63**(1), 65–72 (2008)
16. Zhou, J., Zhang, M., Liu, G., Li, S.: Fuzzy control for automatic steering and line tracking of agricultural robot. Int. Conf. Comput. Sci. Softw. Eng. 1094–1097 (2008)
17. Prema, K., Kumar, N.S., Dash, S.S., Chowdary, S.: Online control of remote operated agricultural robot using Fuzzy controller and virtual instrumentation. Int. Conf. Adv. Eng. Sci. Manag. 196–201 (2012)
18. de Sousa, R.V., Lopes, W.C., Pereira, R.R.D., Inamasu, R.Y., Porto, A.J.V.: A methodology for composing and coordinating primitive Fuzzy behaviors to guide mobile agricultural robots. In: 9th International Conference on Control and Automation, pp. 280–285. IEEE (2011)
19. Zhao, C.J., Jiang, G.Q.: Baseline detection and matching to vision-based navigation of agricultural robot. Int. Conf. Wavelet Anal. Pattern Recogn. 44–48 (2010)
20. Zhang, Y., Gao, F., Tian, L.: INS/GPS integrated navigation for wheeled agricultural robot based on sigma-point Kalman filter. In: 7th International Conference on System Simulation and Scientific Computing, pp. 1425–1431. Asia Simulation Conference (2008)
21. Thamrin, N.M., Arshad, N.H.M., Adnan, R., Sam, R., Razak, N.A., Misnan, M.F., Mahmud, S.F.: Tree detection profile using a single non-intrusive ultrasonic sensor for inter-row tracking application in agriculture field. In: 9th international colloquium on signal processing and its applications, pp. 310–313. IEEE (2013)

22. Xue, J., Xu, L.: Autonomous agricultural robot and its row guidance. In: International Conference on Measuring Technology and Mechatronics Automation, pp. 725–729 (2010)
23. Liu, P., Bi, S., Zang, G., Wang, W., Gao, Y., Deng, Z.: Obstacle avoidance system for agricultural robots based on multi-sensor information fusion. In: International Conference on Computer Science and Network Technology, pp. 1181–1185 (2011)
24. Srivastava, A., Vijay, S., Negi, A., Shrivastava, P., Singh, A.: DTMF based intelligent farming robotic vehicle: An ease to farmers. In: International Conference on Embedded Systems (ICES), pp. 206–210, Coimbatore (2014)
25. Agarwal, N., Thakur, R.: Agricultural robot: Intelligent robot for farming. Int. Adv. Res. J. Sci. Eng. Technol. (IARJSET) **3**(8), 1–5 (2016)
26. Srilekha, K., Monika, J.: Design and operation of Wi-Fi Agribot integrated system. Int. J. Sci. Eng. Technol. Res. (IJSETR) **5**(22), 4473–4478 (2016)
27. Phanomchoeng, G., Saadi, M., Sasithong, P., Tangmongkhonsuk, J., Wijayasekara, S.K., Wuttisittikulkij, L.: Hardware software co-design of a farming robot. Eng. J. **24**(1), 1–10 (2020)
28. Jiang, L., Zhang, Y.: Making agriculture more intelligent: Progress of agricultural robots. Rob. Autom. Eng. J. **4**(1), 1–7 (2018)
29. Agriculture Robots. Available Online: https://www.postscapes.com/agriculture-robots/#indoor-ag-robots
30. Mikhaylov, M.N., Lositskii, I.A.: Control and navigation of forest robot. In: 25th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS), pp. 1–2, St. Petersburg (2018)
31. Hellström, T.: Autonomous navigation for forest machines. A project pre-study in the Department of Computer Science, Umea University, Sweden (2002)
32. Nagasaka, Y., Umeda, N., Kanetai, Y., Taniwaki, K., Sasaki, Y.: Autonomous guidance for rice transplanting using global positioning and gyroscopes. Comput. Electron. Agric. **43**(3), 223–234 (2004)
33. Hellström, T., Johansson, T., Ringdahl, O.: Development of an autonomous forest machine for path tracking. In: P. Corke, S. Sukkariah (eds.) Field and Service Robotics, Springer Tracts in Advanced Robotics, vol. 25, Springer, Berlin, Heidelberg (2006)
34. Li, M., Imou, K., Wakabayashi, K., Yokoyama, S.: Review of research on agricultural vehicle autonomous guidance. Int. J. Agric. Biol. Eng. **2**(3), 1–16 (2008)
35. Ringdahl, O., Lindroos, O., Hellström, T., Bergström, D., Athanassiadis, D., Nordfjell, T.: Path tracking in forest terrain by an autonomous forwarder. Scand. J. For. Res. **26**(4), 350–359 (2011)
36. Gottschalk, R., Burgos-Artizzu, X.P., Ribeiro, A., Pajares, G.: Real-time image processing for the guidance of a small agricultural field inspection vehicle. Int. J. Intell. Syst. Technol. Appl. **8**(1–4), 434–443 (2010)
37. Wang, M., Wang, X., Yi, X., Tu, J.: Experimental study on long-range navigation behavior of agricultural robots. Int. Conf. Comput. Measure. Control Sens. Netw. 409–412 (2012)
38. Ayala, M., Soria, C., Carelli, R.: Visual servo control of a mobile robot in agriculture environments. Mech. Based Des. Struct. Mach. **36**(4), 392–410 (2008)
39. Gao, F., Xun, Y. Wu, J. Bao, G., Tan, Y.: Navigation line detection based on robotic vision in natural vegetation-embraced environment. In: 3rd International Congress on Image and Signal Processing (CISP), pp. 2596–260 (2010)
40. Torres-Sospedra, J., Nebot, P.: A new approach to visual-based sensory system for navigation into orange groves. Sensors **11**(4), 4086–4103 (2011)
41. Zhang, J., Kantor, G., Bergerman, M., Singh, S.: Monocular visual navigation of an autonomous vehicle in natural scene corridor-like environments. In: International Conference on Intelligent Robots and Systems (IROS), pp. 3659–3666, IEEE/RSJ, Vilamoura, Algarve, Portugal (2012)
42. Weiss, U., Biber, P.: Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. Rob. Auton. Syst. **59**, 265–273 (2011)

43. Ball, D., Upcroft, B., Wyeth, G., Corke, P., English, A., Ross, P., Patten, T., Fitch, R., Sukkarieh, S., Bate, A.: Vision-based obstacle detection and navigation for an agricultural robot. J. Field Rob. **33**, 1107–1130 (2016)
44. Han, S., Zhang, Q., Noh, H.: Kalman filtering of DGPS positions for a parallel tracking application. Trans. Am. Soc. Agric. Eng. **45**(3), 553–560 (2002)
45. Nørremark, M., Griepentrog, H.W., Nielsen, J., Søgaard, H.T.: The development and assessment of the accuracy of an autonomous GPS-based system for intra-row mechanical weed control in row crops. Biosys. Eng. **101**(4), 396–410 (2008)
46. Libby, J., Kantor, G.: Deployment of a point and line feature localization system for an outdoor agriculture vehicle. International Conference on Robotics and Automation (ICRA), pp. 1565–1570. IEEE, Shanghai, China (2011)
47. Christiansen, M.: Localization in Orchards using extended Kalman filter for sensor-fusion. Master thesis, University of Southern Denmark (2011)
48. Dørum, J.: Autonomous navigation and row detection in crop fields using computer vision. Master thesis, Norwegian University of Science and Technology (2015)
49. Boubertakh, H., Tadjine, M., Glorennec, P., Labiod, S.: A simple goal seeking navigation method for a mobile robot using human sense, fuzzy logic and reinforcement learning. J. Autom. Control **18**(1), 23–27 (2008)
50. Yousfi, N., Rekik, C., Jallouli, M., Derbel, N.: Optimized Fuzzy controller for mobile robot navigation in a cluttered environment. In: 7th International Multi-Conference on Systems, Signals and Devices, pp. 1–7, IEEE, Amman, Jordan (2010)
51. Troyer, T.A., Pitla, S., Nutter, E.: Inter-row robot navigation using 1D ranging sensors. IFAC-PapersOnLine **49**(16), 463–468 (2016)
52. Gavrilov, A. V., Lee, S.: An architecture of hybrid neural network based navigation system for mobile robot. In: 7th International Conference on Intelligent Systems Design and Applications, pp. 587–590, IEEE, Brazil (2007)
53. Ryerson, A., Zhang, Q.: Vehicle path planning for complete field coverage using genetic algorithms. In: Proceedings of the Automation Technology for Off-road Equipment (ATOE), pp. 309–317, Boon, Germany (2007)
54. Nichols, E., McDaid, L.J., Siddique, N.: Biologically inspired SNN for robot control. IEEE Trans. Cybern. **15**(2), 115–128 (2013)
55. Motlagh, O., Nakhaeinia, D., Tang, S.H., Karasfi, B., Khaksar, W.: Automatic navigation of mobile robots in unknown environments. Neural Comput. Appl. **24**(7), 1569–1581 (2014)
56. Nishiura, Y., Honami, N., Taira, T.: Development of a new grafting method, 4: Robotization of grafting operation. J. Jpn. Soc. Agric. Mach. **61**(6), 103–112 (1999)
57. Osamu, Y., Akiko, F.: Growth adjustment technique of seedling optimal for grafting by seedling storage for cucumber full-automatic machine grafting. Tokyo Agric. Res. **55**(1), 201–202 (2002)
58. Ken, K., Kenta, S., Sadao, S.: Study on automation of seedlings feeding for grafting robot for cucurbitaceous vegetables (Part1). J. Jpn. Soc. Agric. Mach. **68**(6), 117–123 (2006)
59. Kang, C.-H., Han, G.-S., Noh, T.-H., Choi, H.-G.: Splice Grafting Robot for Fruit and Vegetable Plants. World Intellectual Property Organization, WO/2005/089532 (2005)
60. Chen, S., Chiu, Y.C., Chang, Y.C.: Development of a tubing-grafting robotic system for fruit-bearing vegetable seedlings. Appl. Eng. Agric. **26**(4), 707–714 (2010)
61. Lee, J.-M., Kubota, C., Tsao, S.J., Bie, Z.-L., Echevarría, P., Morra, L., Oda, M.: Current status of vegetable grafting: Diffusion, grafting techniques, automation. Sci. Hortic. **127**(2), 93–105 (2010)
62. Chang, Y.-C., Chen, S., Chiu, Y.-C., Lin, L.-H., Chang, Y.-S.: Growth and union acclimation process of sweet pepper grafted by a tubing-grafting robotic system. Hortic. Environ. Biotechnol. **53**(2), 93–101 (2012)
63. Libin, Z., Qinghua, Y., Guanjun, B., Yan, W., Liyong, Q., Feng, G., Fang, X.: Overview of research on agricultural robots in China. Int. J. Agric. Biol. Eng. **1**(1), 12–21 (2008)
64. Kubota, C., McClure, M.A., Kokalis-Burelle, N., Bausher, M.G., Rosskopf, E.N.: Vegetable grafting: History, use, and current technology status in North America. HortScience **43**(6), 1664–1669 (2008)

65. Li, M., Dai, S., Tang, C., Xiang, Y.: Simulation test on plant let-cutting mechanism of grafting robot. Trans. Chin. Soc. Agric. Eng. **24**(6), 129–132 (2008)
66. Zhao, Y., Zhang, T., Wang, H.: Cutting mechanism of root parental stock in automatic sapling grafting machine. Trans. Chin. Soc. Agric. Eng. **24**(9), 79–83 (2008)
67. Jiang, X.: Design of model PJJ-50 grape grafting machine. Agric. Equip. Vehicle Eng. **11**, 7–9 (2011)
68. Wang, F., Liu, M., Wu, X., Cao, X.: Study and design of grafting robot for camellia oleifera seedlings. Forest Machine and Equipment **39**(4), 36–39 (2011)
69. Nezhad, B., Massah, J., Ebrahimpour-Komleh, H.: Design and construction of intelligent tomato picking machine vision. Majlesi J. Electr. Eng (2012)
70. Feng, Q., Wang, X., Wang, G., Li, Z.: Design and test of tomatoes harvesting robot. In: International Conference on Information and Automation, pp. 949–952. IEEE, Lijiang (2015)
71. Xiong, Y., From, P.J., Isler, V.: Design and evaluation of a novel cable-driven gripper with perception capabilities for strawberry picking robots. In: International Conference on Robotics and Automation (ICRA), pp. 7384–739. IEEE, Brisbane, QLD (2018)
72. Ashwini, K.: Survey paper on fruit picking robots. Int. J. Comput. Sci. Mobile Comput. **5**(1), 96–101 (2016)
73. Slaughter, D.C., Giles, D.K., Downey, D.: Autonomous robotic weed control systems: A review. Comput. Electron. Agric. **61**(1), 63–78 (2008)
74. Bakker, T., van K.A., Bontsema, J., Müller, J., van, G.S.: Systematic design of an autonomous platform for robotic weeding. J. Terramech. **47**(2), 63–73 (2010)
75. Pérez-Ruíz, M., Slaughter, D.C., Fathallah, F.A., Gliever, C.J., Miller, B.J.: Co-Robotic intra-row weed control system. Biosys. Eng. **126**, 45–55 (2014)
76. Fennimore, S.A., Slaughter, D.C., Siemens, M.C., Leon, R.G., Saber, M.N.: Technology for automation of weed control in specialty crops. Weed Technol. **30**(4), 823–837 (2016)
77. Bechar, A., Vigneault, C.: Agricultural robots for field operations. Part 2: Operations and systems. Biosys. Eng. **153**, 110–128 (2017)
78. Kunz, C., Weber, J.F., Peteinatos, G.G., Sokefeld, M., Gerhards, R.: Camera steered mechanical weed control in sugar beet, maize and soybean. Precision Agric. **19**(4), 708–720 (2018)
79. Chang, C.-L., Lin, K.-M.: Smart agricultural machine with a computer vision-based weeding and variable rate irrigation scheme. Robotics **7**(3), 38 (2018)
80. Steward, B.L., Gai, J., Tang, L.: The Use of Agricultural Robots in Weed Management and Control. Robotics and Automation for Improving Agriculture, pp. 161–186. Biosystems Engineering Publications (2019)
81. Wu, X., Aravecchia, S., Lottes, P., Stachniss, C., Pradalier, C.: Robotic weed control using automated weed and crop classification. J. Field Robot. **37**(1), 1–29 (2020)
82. Sankaran, S., Mishra, A., Ehsani, R., Davis, C.: A review of advanced techniques for detecting plant diseases. Comput. Electron. Agric. **72**(1), 1–13 (2010)
83. Hengstler, S., Prashanth, D., Fong, S., Hamid, A.: Mesh: A hybrid-resolution smart camera mote for applications in distributed intelligent surveillance. In: 6th International Conference on Information Processing in Sensor Networks, pp. 360–369. New York, USA (2007)
84. Laothawornkitkul, J., Moore, J.P., Taylor, J.E., Possell, M., Gibson, T.D., Hewitt, C.N., Paul, N.D.: Discrimination of plant volatile signatures by an electronic nose: A potential technology for plant pest and disease monitoring. Environ. Sci. Technol. **42**(22), 8433–8439 (2008)
85. Li, Y., Xia, C., Lee, J.: Vision-based pest detection and automatic spray of greenhouse plant. In: International Symposium on Industrial Electronics, pp. 920–925. IEEE, Seoul (2009)
86. Camargoa, A., Smith, J.S.: An image-processing based algorithm to automatically identify plant disease visual symptoms. Biosys. Eng. **102**(1), 9–21 (2009)
87. López, O., Rach, M.M., Migallon, H., Malumbres, M.P., Bonastre, A., Serrano, J.J.: Monitoring pest insect traps by means of low-power image sensor technologies. Sensors **12**(11), 15801–15819 (2012)
88. Francis, J., Anto Sahaya Dhas, D., Anoop, B.K.: Identification of leaf diseases in pepper plants using soft computing techniques. In: Conference on Emerging Devices and Smart Systems (ICEDSS), pp. 168–173. Namakkal (2016)

89. Yazgaç, B.G., Kirci, M., Kivan, M.: Detection of Sunn pests using sound signal processing methods. In: 5th International Conference on Agro-Geoinformatics (Agro-Geoinformatics), pp. 1–6. Tianjin (2016)
90. Gonzalez-de-Santos, P., Ribeiro, A., Fernandez-Quintanilla, C., Lopez-Granados, F., Brandstoetter, M., Tomic, S., Pedrazzi, S., Peruzzi, A., Pajares, G., Kaplanis, G., Perez-Ruiz, M., Valero, C., del Cerro, J., Vieri, M., Rabatel, G., Debilde, B.: Fleets of robots for environmentally-safe pest control in agriculture. Precision Agric. **18**, 574–614 (2017)
91. Blackmore, S., Stout, B., Wang, M., Runov, B.: Robotic agriculture—The future of agricultural mechanization. In: 5th European Conference on Precision Agriculture (ECPA), pp. 621–628. Upsala, Sweden (2005)
92. Jian-sheng, P.: An intelligent robot system for spraying pesticides. Open Electr. Electron. Eng. J. **8**(1), 435–444 (2014)
93. Pilli, S.K., Nallathambi, B., George, S. J., Diwanji, V.: eAGROBOT—A robot for early crop disease detection using image processing. In: 2nd International Conference on Electronics and Communication Systems (ICECS), pp. 1684–1689. Coimbatore (2015)
94. Sharma, S., Borse, R.: Automatic agriculture spraying robot with smart decision making. In: Corchado J., Rodriguez, S., Mitra, S., Thampi, E., El-Alfy, S. (eds) Intelligent Systems Technologies and Applications (ISTA), Advances in Intelligent Systems and Computing, vol. 530, pp. 743–758. Springer, Cham (2016)
95. Sudha, B., Bhuvana, L., Divya, V., Mamathashree, S.R., Pallavi: Automated pest detection and pesticide spraying robot. Int. J. Recent Trends Eng. Res. (IJRTER) **4**(4), 1–8 (2018)
96. Chaitanya, P., Kotte, D., Srinath, A., Kalyan, K.B.: Development of smart pesticide spraying robot. Int. J. Recent Technol. Engi. (IJRTE) **8**(5), 2193–2202 (2020)
97. Ceres, R., Pons, J.L., Jiménez, A.R., Martín, J.M., Calderón, L.: Design and implementation of an aided fruit-harvesting robot (Agribot). Industr. Robot: An Int. J. **25**(5), 337–346 (1998)
98. Hua, Y., Zhang, N., Yuan, X., Quan, L., Yang, J., Nagasaka, K., Zhou, X.-G.: Recent advances in intelligent automated fruit harvesting robots. Open Agric. J. **13**(1), 101–106 (2019)
99. Onishi, Y., Yoshida, T., Kurita, H., Fukao, T., Arihara, H., Iwai, A.: An automated fruit harvesting robot by using deep learning. ROBOMECH J. **6**(1), 1–8 (2019)
100. Yuan, Y., Zhang, X., Zhao, H.: Apple harvesting robot picking path planning and simulation. In: International Conference on Information Engineering and Computer Science, pp. 1–4, Wuhan (2009)
101. Lv, J., Zhao, D., Ji, W., Chen, Y., Shen, H.: Design and research on vision system of apple harvesting robot. In: 3rd International Conference on Intelligent Human-Machine Systems and Cybernetics, pp. 177–180, Zhejiang (2011)
102. De-An, Z., Jidong, L., Wei, J., Ying, Z., Yu, C.: Design and control of an apple harvesting robot. Biosys. Eng. **110**(2), 112–122 (2011)
103. Li, Z., Liu, J., Li, P., Li, W.: Analysis of workspace and kinematics for a tomato harvesting robot. In: International Conference on Intelligent Computation Technology and Automation (ICICTA), pp. 823–827, Hunan (2008)
104. Wang, J., Zhou, Z., Du, X.: Design and co-simulation for tomato harvesting robots. In: 31st Chinese Control Conference, pp. 5105–5108, Hefei (2012)
105. Liu, J., Li, Z., Wang, F., Li, P., Xi, N.: Hand-arm coordination for a tomato harvesting robot based on commercial manipulator. In: International Conference on Robotics and Biomimetics (ROBIO), pp. 2715–2720. IEEE, Shenzhen (2013)
106. Yaguchi, H., Nagahama, K., Hasegawa, T., Inaba, M.: Development of an autonomous tomato harvesting robot with rotational plucking gripper. In: International Conference on Intelligent Robots and Systems (IROS), pp. 652–657. IEEE/RSJ, Daejeon, (2016)
107. Wang, G., Yu, Y., Feng, Q.: Design of end-effector for tomato robotic harvesting. IFAC-PapersOnLine **49**(16), 190–193 (2016)
108. Sakai, S., Osuka, K., Fukushima, H., Iida, M.: Watermelon harvesting experiment of a heavy material handling agricultural robot with LQ control. In: International Conference on Intelligent Robots and Systems, vol. 1, pp. 769–774. IEEE/RSJ, Lausanne, Switzerland (2002)

109. Arima, S., Shibusawa, S., Kondo, N., Yamashita, J.: Traceability based on multi-operation robot; Information from spraying, harvesting and grading operation robot. In: International Conference on Advanced Intelligent Mechatronics (AIM 2003), vol. 2, pp. 1204–1209. IEEE/ASME, Kobe, Japan (2003)
110. Qingchun, F., Wengang, Z., Quan, Q., Kai, J., Rui, G.: Study on strawberry robotic harvesting system. In: International Conference on Computer Science and Automation Engineering (CSAE), pp. 320–324. IEEE, Zhangjiajie (2012)
111. Irie, N., Taguchi, N., Horie, T., Ishimatsu, T.: Asparagus harvesting robot coordinated with 3-D vision sensor. In: International Conference on Industrial Technology, pp. 1–6. IEEE, Gippsland, VIC (2009)
112. Birrell, S., Hughes, J., Cai, J.Y., Iida, F.: A field-tested robotic harvesting system for iceberg lettuce. J. Field Rob. **37**(2), 225–245 (2020)
113. Hajjaj, S.S.H., Sahari, K.S.M.: Review of agriculture robotics: Practicality and feasibility. In: International Symposium on Robotics and Intelligent Sensors (IRIS), pp. 194–198. IEEE, Tokyo (2016)
114. Behmanesh, M., Hong, T.S., Kassim, M.S.M., Azim, A., Dashtizadeh, Z.: A brief survey on agricultural robots. Int. J. Mech. Eng. Rob. Res. (IJMERR) **6**(3), 178–182 (2017)
115. Shamshiri, R.R., Weltzien, C., Hameed, I.A., Yule, I.J., Grift, T.E., Balasundram, S.K., Pitonakova, L., Ahmad, D., Chowdhary, G.: Research and development in agricultural robotics: A perspective of digital farming. Int. J. Agric. Biol. Eng. (IJABE) **11**(4), 1–14 (2018)
116. Jadhav, P.K., Deshmukh, S.S., Khairnar, P.N.: Survey paper on AgRo-bot autonomous robot. Int. Res. J. Eng. Technol. (IRJET) **6**(12), 434–441 (2019)
117. Fue, K.G., Porter, W.M., Barnes, E.M., Rains, G.C.: An extensive review of mobile agricultural robotics for field operations: Focus on cotton harvesting. AgriEngineering **2**(1), 150–174 (2020)
118. Veroustraete, F.: The rise of the drones in agriculture. Ecronicon **2**(2), 1–3 (2015)
119. Natu, A.S., Kulkarni, S.C.: Adoption and utilization of drones for advanced precision farming: A review. Int. J. Recent Innov. Trends Comput. Commun. **4**(5), 563–565 (2016)
120. Ahirwar, S., Swarnkar, R., Bhukya, S., Namwade, G.: Application of drone in agriculture. Int. J. Curr. Microbiol. Appl. Sci. **8**(1), 2500–2505 (2019)
121. Mogili, U.R., Deepak, B.: Review on application of drone systems in precision agriculture. Proc. Comput. Sci. **133**, 502–509 (2018)
122. Daponte, P., De Vito, L., Glielmo, L., Iannelli, L., Liuzza, D., Picariello, F., Silano, G.: A review on the use of drones for precision agriculture. IOP Conf. Ser. Earth Environ. Sci. **275**, 1–11 (2019)
123. Abdullahi, H.S., Mahieddine, F., Sheriff, R.E.: Technology impact on agricultural productivity: A review of precision agriculture using unmanned aerial vehicles. In: Pillai, P., Hu, Y., Otung, I., Giambene, G. (eds) Wireless and Satellite Systems, WiSATS, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 154, pp. 388–400. Springer, Cham (2015)
124. Primicerio, J., Di Gennaro, S.F., Fiorillo, E., Genesio, L., Lugato, E., Matese, A., Vaccari, F.P.: A flexible unmanned aerial vehicle for precision agriculture. Precis. Agric. **13**(4), 517–523 (2012)
125. Bendig, J., Bolten, A., Bareth, G.: Introducing a low-cost mini-UAV for thermal-and multispectral-imaging. International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XXII ISPRS Congress, vol. XXXIX-B1, pp. 345–349, Melbourne, Australia (2012)
126. Anthony, D., Elbaum, S., Lorenz, A., Detweiler, C.: On crop height estimation with UAVs. In: International Conference on Intelligent Robots and Systems, pp. 4805–4812. IEEE/RSJ, Chicago, IL (2014)
127. Huang, Y., Hoffmann, W.C., Lan, Y., Wu, W., Fritz, B.K.: Development of a spray system for an unmanned aerial vehicle platform. Appl. Eng. Agric. **25**(6), 803–809 (2009)
128. Faiçal, B.S., Costa, F.G., Pessin, G., Ueyama, J., Freitas, H., Colombo, A., Fini, P.H., Villas, L., Osório, F.S., Vargas, P.A., Braun, T.: The use of unmanned aerial vehicles and wireless sensor networks for spraying pesticides. J. Syst. Architect. **60**(4), 393–404 (2014)

129. Faiçal, B.S., Freitas, H., Gomes, P.H., Mano, L.Y., Pessin, G., de Carvalho, A.C.P.L.F., Krishnamachar, B., Ueyama, J.: An adaptive approach for UAV-based pesticide spraying in dynamic environments. Comput. Electron. Agric. **138**, 210–223 (2017)
130. Kurkute, S.R., Deore, B.D., Kasar, P., Bhamare, M., Sahane, M.: Drones for smart agriculture: A technical report. Int. J. Res. Appl. Sci. Eng. Technol. (IJRASET) **6**(IV), 341–346 (2018)
131. Talaviya, T., Shah, D., Patel, N., Yagnik, H., Shah, M.: Implementation of artificial intelligence in agriculture for optimisation of irrigation and application of pesticides and herbicides. Artif. Intell. Agric. **4**, 58–73 (2020)
132. Thompson, L.J., Shi, Y., Ferguson, R.B.: Getting Started with Drones in Agriculture (G-2296). University of Nebraska Extension, NebGuide (2017)
133. Zheng, H., Zhou, X., Cheng, T., Yao, X., Tian, Y., Cao, W., Zhu, Y.: Evaluation of a UAV based hyperspectral frame camera for monitoring the leaf nitrogen concentration in rice. In: International Geoscience and Remote Sensing Symposium (IGARSS), pp. 7350–7353. IEEE, Beijing (2016)
134. Reinecke, M., Prinsloo, T.: The influence of drone monitoring on crop health and harvest size. In: 1st International Conference in Next Generation Computing Applications (NextComp), pp. 5–10. IEEE, Mauritius (2017)
135. Psirofonia, P., Samaritakis, V., Eliopoulos, P., Potamitis, I.: Use of unmanned aerial vehicles for agricultural applications with emphasis on crop protection: Three novel case-studies. Int. J. Agric. Sci. Technol. **5**(1), 30–39 (2017)
136. Yallappa, D., Veerangouda, M., Maski, D., Palled, V., Bheemanna, M.: Development and evaluation of drone mounted sprayer for pesticide applications to crops. Glob. Humanitarian Technol. Conf. (GHTC), pp. 1–7. IEEE, San Jose, CA (2017)
137. Pharne, I.D., Kanase, S., Patwegar, S., Patil, P., Pore, A., Kadam, Y.: Agriculture drone sprayer. Int. J. Recent Trends Eng. Res. **4**(3), 181–185 (2018)
138. Xiang, H., Tian, L.: Development of a low-cost agricultural remote sensing system based on an autonomous unmanned aerial vehicle (UAV). Biosys. Eng. **108**(2), 174–190 (2011)
139. Senthilnath, J., Dokania, A., Kandukuri, M.K.N.R., Anand, G., Omkar, S.N.: Detection of tomatoes using spectral-spatial methods in remotely sensed RGB images captured by UAV. Biosys. Eng. **146**, 16–36 (2016)
140. Kushwaha, H.L., Sinha, J.P., Khura, T., Kushwaha, D.K.: Status and scope of robotics in agriculture. Int. Conf. Emerg. Technol. Agric. Food Eng. 264–277 (2016)
141. Roldán, J.J., del Cerro, J., Garzón-Ramos, D., Garcia-Aunon, P., Garzón, M., de León, J., Barrientos, A.: Robots in agriculture: State of art and practical experiences. Appl. Rob. Greenhouse Farming 67–90 (2018)
142. Emmi, L., de Soto, M.G., Pajares, G., de Santos, P.G.: New trends in robotics for agriculture: Integration and assessment of a real fleet of robots. Sci. World J. **2014**, 21 (2014)

Detection of Rotten Fruits and Vegetables Using Deep Learning

Susovan Jana, Ranjan Parekh, and Bijan Sarkar

1 Introduction

Fruits and vegetables are very necessary items for our daily life. There are different species of edible fruits and vegetables in nature. Fresh fruits and vegetables are not only delicious to eat but also a good source of many important vitamins or minerals. Fresh fruits and vegetables are used in the food processing industries to process delicious food products. The fruits and vegetables have to pass through various stages from harvesting to reach the customer. The stages are harvesting, sorting, classification, grading, etc. The manual execution of those tasks requires lots of expert resources and a long time. Many countries are suffering from a resource shortage for agricultural tasks because of a lack of interest in such a laborious job. Hence, automation is needed in every aspect of the processing of fruits and vegetables. Computer vision and machine learning have earned huge success in solving various automation problems in different industries. The researchers also contributed to addressing various problems in fruits and vegetable processing with the help of computer vision and machine learning techniques. This chapter explores those problems and challenges of fruits and vegetable processing using computer vision and machine learning techniques. The major focus has been given on the problem of automatic detection of rotten fruits and vegetables.

S. Jana (✉) · R. Parekh
School of Education Technology, Jadavpur University, Kolkata 700032, India
e-mail: jana.susovan2@gmail.com

R. Parekh
e-mail: rparekh.edutech@jadavpuruniversity.in

B. Sarkar
Department of Production Engineering, Jadavpur University, Kolkata 700032, India
e-mail: bijan.sarkar@jadavpuruniversity.in

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_3

Most of the time, the shape, color, and texture are changed on the surface of rotten fruit and vegetable. The bad smell is also an important indication of rot. The fruits and vegetables mostly rot in the inventory. There are many factors for the fruit or vegetable to become rotten [1, 2]. The factors are temperature, moisture, air, light, and microorganisms. The fruit and vegetables also rot during transportation [3, 4]. A single rotten fruit or vegetable can damage multiple fresh fruit and vegetable in inventory. Inventory damage causes a good amount of loss in the business of fruits and vegetables. The early detection of rotten fruits and vegetables reduces the amount of damage inside inventory or store and also enhances food safety. Manual resource detects rotten fruits and vegetables by smelling, observing the shape deformation, and change in surface color, and texture. The smell cannot be tested in case of automatic detection of rotten fruits and vegetables using computer vision and machine learning. The computer vision has to rely only on the change of surface feature compared with the fresh one. It makes the task of computer-based detection of rotten fruits and vegetables into a challenging task for researchers. This chapter addressed the problem of rotten fruit and vegetable detection using state-of-the-art deep learning techniques. A convolutional neural network (CNN) architecture has been proposed to classify the rotten and fresh from a captured image of fruit and vegetable.

This chapter has been structured as follows: Sect. 2 describes the state-of-the-art problems and challenges of fruits and vegetable processing using computer vision and machine learning techniques. Section 3 elucidates the materials and the proposed method in detail. Section 4 brings experiments and results. A detailed discussion on this work has been presented in Sect. 5. Section 6 concludes the chapter with future scope.

2 Computer Vision and Machine Learning in Fruits and Vegetable Processing

The computer vision and machine learning had already achieved astounding success in many automation challenges regarding fruits and vegetable processing. Computer vision completely relies on the appearance of the outer surface of fruits or vegetables. The literature on fruits and vegetable processing can be broadly categorized based on problems. This section highlights some of the very challenging problems of fruits and vegetable processing i.e. segmentation and detection of fruits and vegetables from the natural environment, classification of fruits and vegetable type, grading the fruits and vegetables, sorting the defective fruits, and vegetables.

2.1 Segmentation and Detection of Fruits and Vegetables from the Natural Environment

The object segmentation is a very common problem in the domain of computer vision. The task of fruit and vegetable segmentation becomes tedious when the background is a natural environment. The natural background is very complex because it contains leaves, stem, sky, etc. [5]. The segmentation of fruits and vegetables is a preliminary step for on tree detection of fruits and vegetables. The fruits and vegetables are segmented using the color properties in different color spaces [6]. The segmented object region has been passed through different morphological operations [7] to refine the object region. Most of the time different edge detection [8] techniques are applied for boundary contour extraction. The Hough transform [9] or circle regression [10] techniques are applied to detect actual fruit or vegetable region from the boundary contour. The deep learning models can also be used for the detection of fruits and vegetables from the natural environment [11]. There are lots of scopes for improvements. The challenges are (a) partial occlusion by leaves or branches (b) overlapping similar fruits and vegetables (c) the color of fruit or vegetable object is similar to the background e.g. the green fruit or vegetable with green leaf.

2.2 Classification of Fruits and Vegetables

The classification problem of fruits and vegetables has been explored a lot in the last two decades. The steps, which are followed by the majority of the researchers for fruits and vegetable classification, are pre-processing, feature extraction, train a supervised model, and predict the class for unknown fruits and vegetable samples by this trained model. The preprocessing steps include binarization, morphological operations, noise removal, etc. The visual features for classification are shape [12], color [13], and texture [14]. The popular shape and size features are region area, perimeter, major axis length, minor axis length, roundness [15], etc. The commonly used texture features for fruits and vegetables classification are the statistical descriptor from GLCM [15], histogram oriented gradient (HOG), local binary pattern (LBP), and Gabor wavelet [16], etc. The color features can be histogram [17], and mean, standard deviation, skewness, kurtosis [18] of different color channels in different color spaces. The frequently used conventional machine learning models [19] for classification are Naïve Bayes [12], kNN [17], Random Forest [18], Linear Discriminant Analysis [14, 19], Support Vector Machine [15], and Neural Network [20], etc. The state of the art deep learning techniques is also applied to address this problem [21]. Still, there are sufficient scopes for improvement. The scopes for future research are (a) intra-class dissimilarities and inter-class similarity (b) change of viewing position and illumination condition (c) change of visual properties in different growth stages.

2.3 Grading of Fruits and Vegetables

The grading of fruits and vegetables is very important for getting an appropriate price at the time of sale. It is also helpful to the different categories of customers. The grading of fruits and vegetables can be done with various parameters. The popular parameters of fruits and vegetable grading are shape [22], maturity [23], volume [24], weight [22], etc. The exact region should be segmented before measuring those parameters. The perfect segmentation leads to accurate grading. The viewing position is a constraint for measuring those parameters. The existing literature proposes grading techniques for mostly the regular shaped fruits and vegetables i.e. spherical [25], elliptical, paraboloid [26], cylindrical [22], and axisymmetric [27] fruits and vegetables. The grading of irregular and non-axisymmetric fruits and vegetables could be a very good scope for further research.

2.4 Sorting the Defective Fruits and Vegetables

This chapter is mainly focused on the sorting of rotten fruits and vegetables. Hence, a detailed survey has been presented for this problem. Chandini et al. proposed a technique for the detection of fresh and defective apple [28]. Authors considered two types of defects in apple fruit i.e. rot and scab. At first, the RGB input image was converted to HIS color space. Then k-means clustering was used to segment the defective region. Contrast, correlation, energy, and homogeneity were extracted from the Gray level co-occurrence matrix (GLCM) and fed into a multiclass support vector machine (SVM) classifier. The SVM classifier did the prediction among fresh, rot, and scab for the unknown samples. They were able to reach 85.64% of classification accuracy using their technique. Karakaya et al. proposed a technique to classify rotten and fresh fruit [29]. The input images were segmented using the Otsu segmentation technique. The extracted features from the segmented image were histogram, GLCM, and Bag of Features. The authors had experimented with those features on 1200 images. The images were collected from a public dataset. The SVM classifier was used in experimentation with 10-fold cross-validation and RBF kernel. Yogesh et al. proposed a computer vision-based system for detecting the defective and non-defective fruit [30]. The system also classified the stage of the defect after detecting the defect in a fruit. A dataset of 1200 images was collected. The dataset contains images of RGB color format. The images were pre-processed and segmented from the background. The extracted features were the number of objects, connectivity, area, perimeter, major axis, minor axis, convex area, diameter, eccentricity, filled area, solidity, and Euler number. The SVM classifier was able to detect the defective fruits with the stage of defect more accurately than that of kNN and AlexNet. The attack of Penicillium fungi is a reason for the rot of citrus fruit. Previously, those fungi affected and rotten citrus fruit was detected manually with the help of ultraviolet rays. It was very harmful to manual resources. Gómez-Sanchis et al. proposed a machine learning-based approach to detect the rotten citrus fruit caused by Penicillium fungi [31]. A dataset of hyperspectral images was formed as a part of that research. The extracted features from those images were citriculture, 114 spatio-spectral features, and 57 spectral features. The detection accuracy using artificial neural networks (ANN) was maximum among all the classifiers used for the same purpose. Kamalakannan et al. proposed a defect detection and classification system for mandarin fruit using image analysis [32]. The authors had used a fuzzy segmentation technique. A binary wavelet transform (BWT) was chosen as a classification feature. A rule-based linear classifier was used to do the final classification using the extracted feature. Capizzi et al. also proposed a defect detection and classification technique for orange fruits using surface features [33]. HSV histogram and GLCM features were extracted to classify the defect of orange. The Radial Basis Probabilistic Neural Network does the task of the classification. Another classification system for separating diseased and non-diseased fruits was proposed by Ranjit et al. [34]. At first, the defective region was segmented by k-means clustering. Then the shape, color, and texture features were extracted for classification with the help of the SVM classifier. The mixture of visual and non-visual features was used to determine the freshness index of eggplant [35]. The segmentation rotten region has been explored [36] and a color based clustering technique was proposed by Roy et al. [37].

The machine learning algorithms will be appropriate to detect rotten fruits and vegetables from the lot. The surface appearance helps to detect rotten fruits and vegetables. The changes are visible in surface textures and color from the fresh one. The challenge arises when there is more intra-class dissimilarity e.g. the appearance of rotten fruits and vegetables varies over different fruit and vegetable class. Most of the previous approaches were based on surface texture, histogram, and color features. The prior approaches were proposed to classify fresh and rotten for a specific type of fruit or vegetable. Hence, the proposed technique should be able to detect rotten fruit and vegetable from a lot of similar types of fruits and vegetables as well as from a lot of different varieties of fruits and vegetables. Convolutional neural network architecture is proposed in this work to classify between fresh and rotten fruits and vegetables.

3 Materials and Methods

The proposed method will be very effective for the automatic detection of rotten fruits and vegetables from the lot. The proposed method is completely based on the state of the art deep learning technique. Convolutional neural network architecture is designed here for performing the task of classification into a rotten or fresh category of a fruit or vegetable. The proposed CNN model is trained with the images of fresh as well as rotten fruits and vegetables of various types with the corresponding labels. The trained CNN model will detect the rotten fruits and vegetables from an unknown image.

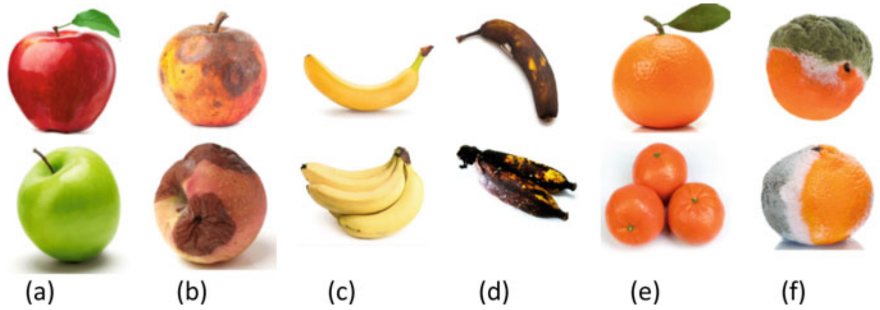


Fig. 1 Samples from dataset—**a** Fresh Apple, **b** Rotten Apple, **c** Fresh Banana, **d** Rotten Banana, **e** Fresh Orange, **f** Rotten Orange

3.1 Dataset

The images were collected from an online source [38] to make the dataset. The images belong from 3 different categories of fruits i.e. apple, banana, and orange. Each of the fruit categories has two classes of images i.e. fresh and rotten. The dataset contains fresh apple (232), rotten apple (327), fresh banana (218), rotten banana (306), fresh orange (206), and rotten orange (222). The dataset introduces a good number of intra-class varieties to enhance the robustness of the model.

Figure 1 shows a few samples from the dataset. The image augmentation technique was applied here to increase the number of images in the dataset. All the samples were rotated in five different directions i.e. $15°, 30°, 45°, 60°, 75°$. The salt and pepper noise was added over all the images. The images were also translated and flipped vertically. In total 8 different data augmentation technique was applied to increase the number as well as the variety in the dataset. The augmented final dataset contains 13,599 images in total.

3.2 Convolutional Neural Network

The convolutional neural network is a very popular deep learning algorithm for image classification, object recognition, etc. The artificial neural network can be used on an image if the image can be converted to a 1D list of pixel intensities. The problem is that the 1D list losses the spatial information of pixels whereas CNN extracts features by preserving spatial information among pixels. A 2D filter convolves through the image to extract various features like curve, edge, colors, etc. The filter size should be large enough to accommodate features containing many pixels as well as small enough it can be used repetitively. Figure 2 shows a demonstration of convolution. Here, the original image is a 6×6 binary image. The convolution filter is a 3×3 matrix. The convolution starts from the top-left corner without padding and stride as 1. Every time the filter is multiplied with the corresponding elements in the image.

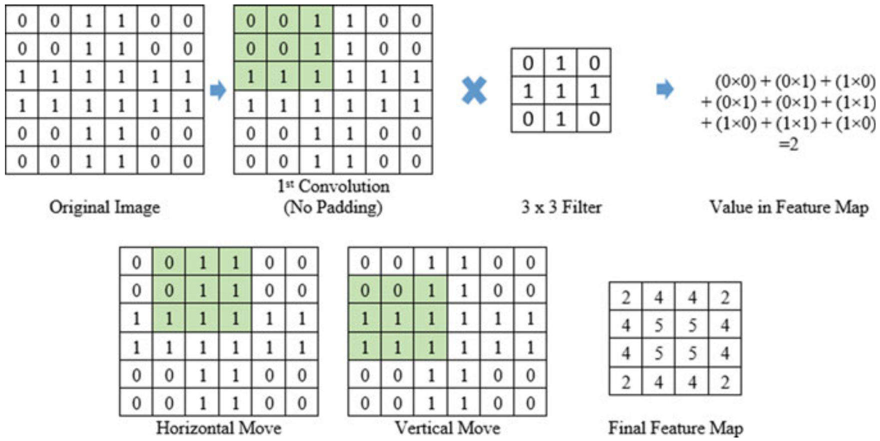


Fig. 2 A simple demonstration of convolution over 2D binary image

The sum of multiplied elements is taken from each move of the filter to generate the feature map. The filter generally moves through the 2D image from left to right and top to bottom. The filter moves separately over different channels for color images containing multiple channels. The reason for using multiple convolution filters is that the different filter extracts different feature maps. The combined feature map improves the classification performance. The stride is the number of pixels to escape in a single move. The larger strides minimize the feature but increase the chance of missing small features. The padding is the process of adding dummy pixels on the different sides of the image to generate the feature map of the same dimension as the image. The Rectified Linear Unit (ReLU) is added very often after extracting a basic feature map to add non-linearity by an activation function for further processing. The dimensionality of feature maps sometimes becomes a headache for a network concerning time as well as processing. Hence, pooling is used to reduce the feature map with minimal information loss. There are different types of pooling i.e. max-pooling (takes pixels with maximum value), average pooling (takes average value of pixels), sum pooling (takes sum of the pixel values), etc. The max-pooling is very popular for image classification problems. Figure 3 shows an example of max pooling. The maximum value from each colored region is picked for max pooling.

3.3 Proposed Convolutional Neural Network Architecture

This chapter overcomes the challenges of rotten fruit and vegetable detection using conventional machine learning models. A convolution neural network architecture has been proposed here. The network architecture is sequential. Figure 4 depicts the detailed architecture of this model. The input layer receives 64 $\times$ 64 RGB color images with zero center normalization. A convolution layer is added next to the

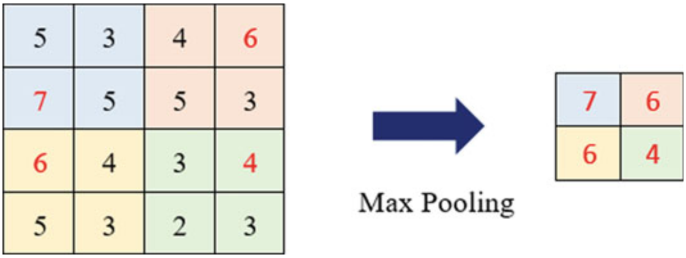


Fig. 3 A simple example of max pooling

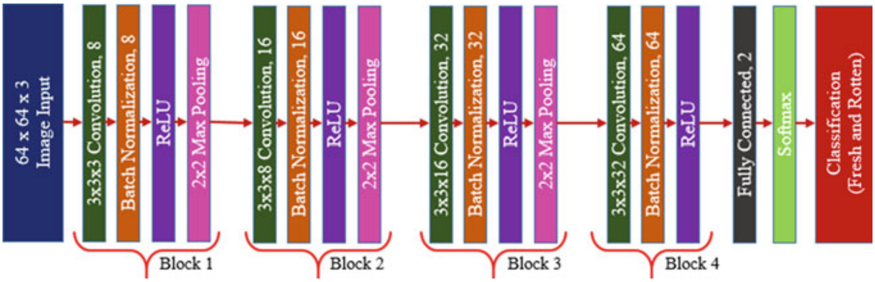


Fig. 4 The architecture of the proposed CNN model

input layer. The layer contains 8 number of 3×3 convolution filters with stride [1 1] and zero paddings. The padding size is set in such a way so that the output layer will have the same size as input. The convolution does the extraction of the features from the input image as long as the training progresses. The features are the discriminating visual features of any fresh or rotten fruit and vegetables. The rotten fruit and vegetable surface color and texture are not continuous. The color and textures of rotten regions change over the image compared with a fresh fruit and vegetable surface. The convolution layer is followed by a batch normalization with 8 channels and a ReLU layer. The batch normalization layer normalizes the features learned from different input layers. It gives the network flexibility of learning independently from different layer and also speed up the training process. The ReLU layer is used to add nonlinearity with a nonlinear activation function. Refer to Eq. (1). A 2×2 max-pooling layer is added next to ReLU layer with stride [2 2] and padding [0 0 0 0]. The first block of the convolution layer, batch normalization layer, ReLU layer, and the max-pooling layer is formed with those parameters.

Another three similar types of blocks are added sequentially one after another. Only the number of filters in the convolution layer and the number of channels in batch normalization layers have been doubled as the new blocks have been added. In the final block, the max-pooling layer is replaced with a fully connected layer. Then a Softmax layer, refer to Eq. (2), is added before the final classification layer. The Softmax layer normalizes the output of the fully connected layer and it produces the probabilities which will be used by the classification layer to predict the class of unknown test sample. The final output is the class label i.e. Fresh or Rotten. The classification layer uses the binary cross-entropy for the loss computation. Refer to Eq. (3). Here, i stands for the number of classes. There are two classes as it is a binary classification problem, $t_1 = 1$ for the positive class and $t_1 = 0$ for the negative class. The loss can be represented as in Eq. (4).

$$f(x) = \begin{Bmatrix} x, x \geq 0 \\ 0, x < 0 \end{Bmatrix} \tag{1}$$

$$f(S)_i = \frac{e^{s_i}}{\sum_{k=1}^{K} e^{z_k}} \tag{2}$$

$$\text{CE} = -\sum_{i=1}^{c=2} t_i \log(f(s_i)) = -t_1 \log(f(s_1)) - (1 - t_1)\log(1 - f(s_1)) \tag{3}$$

$$\text{CE} = \begin{cases} -\log(f(s_1)) & \text{if}\, t_1 = 1 \\ -\log(1 - f(s_1)) & \text{if}\, t_1 = 0 \end{cases} \tag{4}$$

3.4 AlexNet Architecture

AlexNet [39] is a pre-trained convolutional neural network. The architecture of AlexNet has been specially designed for object classification from high-resolution images. It has been trained on 1000 classes of the ImageNet dataset. The model won the second-best position in the ILSVRC-2012 competition. The model takes an input of a uniform size 227 × 227 × 3. The net contains 5 convolution layers, 7 ReLU layers, 2 cross channel normalization layers, 3 max-pooling layers, and 3 fully connected layers. Two dropout layer was included for two fully connected layers to reduce the overfitting. The final fully connected layer of 1000 nodes followed by a softmax layer and a classification layer with a cross-entropy loss function. Transfer learning is a way of using the popular pre-trained network architecture for a customized classification problem. The AlexNet model has been trained millions of images with a wider range of classes. The model has already learned the rich feature representation. Sometimes the fine-tuning of the pre-trained model is easier and faster than training a new model from scratch with random weights. Hence, the transfer leaning is done on a pre-trained AlexNet model to classify a fruit image into a fresh and rotten category. The final three layers have been replaced by the fully connected layer with two nodes i.e. fresh and rotten. A softmax and a classification layer with binary cross-entropy loss function follow the fully connected layer. The detailed architecture of this model is shown in Fig. 5.

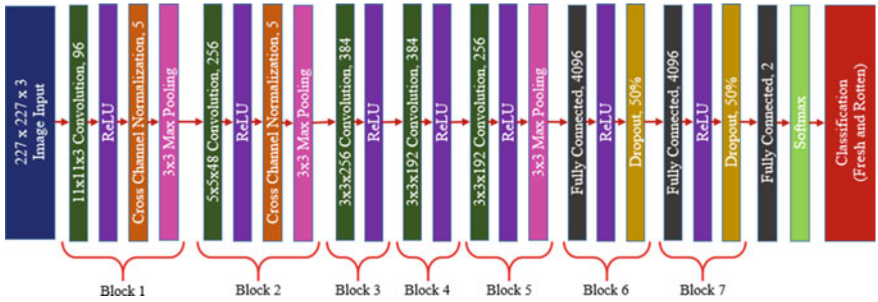


Fig. 5 The architecture of fine-tuned AlexNet using transfer learning

4 Experimentation and Results

The experimentations have been carried out to test the robustness and effectiveness of the proposed CNN model. In total four different sets of images have been created from the actual dataset. Set 1 contains images of two classes i.e. fresh apple and rotten apple. Similarly, set 2 contains images of two classes i.e. fresh banana and rotten banana. Set 3 also contains images of two classes i.e. fresh orange and rotten orange. Set 4 is the complete dataset with two classes i.e. fresh or rotten. The fresh class in Set 4 contains images of fresh fruits of all three types. The rotten class in Set 4 contains images of the rotten fruit of all three types. The classes and distribution of training and testing images for each dataset are mentioned in Table 1. The training and testing data have been chosen randomly from there. The images are resized to 64×64 for the proposed CNN. The fine-tuning of training parameters is very important to build a very robust model. The training data was also shuffled in every epoch. The initial learning rate is 0.01. The maximum number of the epoch is 25 for all datasets. The proposed CNN model has been trained 4 times on dataset 1. Each time the training and testing images are chosen randomly after shuffling the dataset 1. The final result is prepared by averaging the result of four tests on dataset 1. The same is performed

Table 1 Distribution of training and testing images in different datasets

Dataset	Class	Total	Training	Testing
Dataset 1	Fresh Apple	2088	1600	488
	Rotten Apple	2943	1600	1343
Dataset 2	Fresh Banana	1962	1600	362
	Rotten Banana	2754	1600	1154
Dataset 3	Fresh Orange	1854	1600	254
	Rotten Orange	1998	1600	398
Dataset 4	Fresh Fruits	5904	4800	1104
	Rotten Fruits	7695	4800	2895

Table 2 The performance of the proposed CNN model on different datasets

Dataset	Fresh and predicted as fresh (%)	Fresh but predicted as rotten (%)	Rotten but predicted as fresh (%)	Rotten and predicted as rotten (%)	Overall accuracy (%)
Dataset 1	98.51	1.49	1.38	98.62	98.59
Dataset 2	99.93	0.07	0.09	99.91	99.92
Dataset 3	98.52	1.48	1.07	98.93	98.77
Dataset 4	97.60	2.40	2.20	97.80	97.74

with the other three datasets i.e. dataset 2, dataset 3, and dataset 4. Table 2 depicts the average of the four tests on each dataset using the proposed CNN model. Figure 6 shows the training vs validation accuracy of the proposed CNN model on dataset 4. Figure 7 depicts the training vs validation loss for the same.

The images of datasets are resized to 227 × 227 for AlexNet. The AlexNet is also trained by transfer learning with this dataset. The fine-tuned AlexNet is tested separately with each of the datasets as done for the proposed CNN model. The results are shown in Table 3 for all the datasets using the fine-tuned AlexNet model.

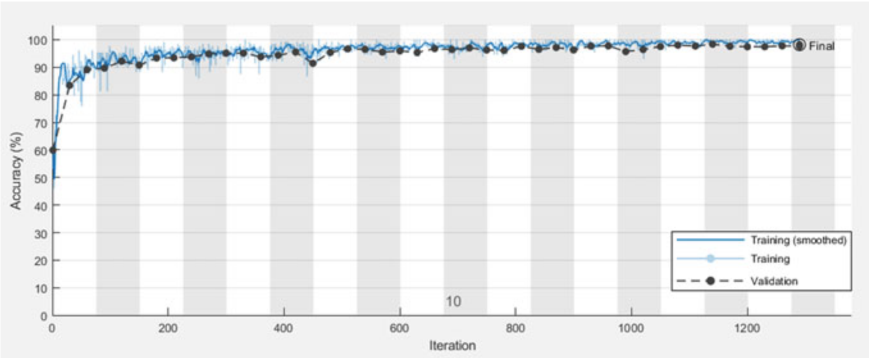


Fig. 6 Training versus validation accuracy of the proposed CNN model on dataset 4

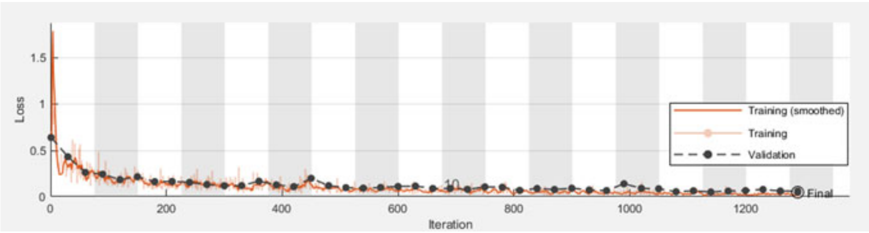


Fig. 7 Training versus validation loss of the proposed CNN model on dataset 4

Table 3 The performance of the fine-tuned AlexNet on different datasets

Dataset	Fresh and predicted as fresh (%)	Fresh but predicted as rotten (%)	Rotten but predicted as fresh (%)	Rotten and predicted as rotten (%)	Overall accuracy (%)
Dataset 1	97.95	2.05	0.34	99.66	99.21
Dataset 2	98.55	1.45	0.00	100.00	99.65
Dataset 3	100.00	0.00	0.31	99.69	99.81
Dataset 4	99.43	0.57	3.11	96.89	97.59

Some previous approaches have been implemented for the classification of fresh and rotten fruits and vegetables and also tested on the same datasets as done in the proposed approach. The histogram features were used by Capizzi et al. [33], and Karakaya et al. [29]. The GLCM based features were used by Karakaya et al. [29], Chandini et al. [28], and Capizzi et al. [33]. Karakaya et al. also used a bag of features [29] for this problem. Most of the approaches used SVM classifier to classify fresh and rotten using those features. Hence, all the features are extracted and used separately to classify fresh and rotten with the help of SVM. The results have been reported in the same way as done for the proposed approach. The same parameters, which are used to represent the performance of the proposed approach, are used here to represent the performance of the previous approaches as well. Tables 4, 5, and 6 depict the average results of four runs with grayscale histogram features, GLCM features, and a bag of features correspondingly.

Table 4 The performance using Grayscale Histogram + SVM on different datasets

Dataset	Fresh and predicted as fresh (%)	Fresh but predicted as rotten (%)	Rotten but predicted as fresh (%)	Rotten and predicted as rotten (%)	Overall accuracy (%)
Dataset 1	48.31	51.69	41.25	58.75	55.97
Dataset 2	95.93	4.07	2.99	97.01	96.75
Dataset 3	53.94	46.06	37.69	62.31	59.05
Dataset 4	53.89	46.11	48.59	51.41	52.09

Table 5 The performance using GLCM features + SVM on different datasets

Dataset	Fresh and predicted as fresh (%)	Fresh but predicted as rotten (%)	Rotten but predicted as fresh (%)	Rotten and predicted as rotten (%)	Overall accuracy (%)
Dataset 1	76.38	23.62	43.17	56.83	62.04
Dataset 2	89.23	10.77	13.26	86.74	87.34
Dataset 3	84.15	15.85	36.18	63.82	71.74
Dataset 4	79.91	20.09	28.99	71.01	73.47

Table 6 The performance using Bag of Features + SVM on different datasets

Dataset	Fresh and predicted as fresh (%)	Fresh but predicted as rotten (%)	Rotten but predicted as fresh (%)	Rotten and predicted as rotten (%)	Overall accuracy (%)
Dataset 1	76.13	23.88	19.34	80.66	78.39
Dataset 2	94.75	5.25	6.30	93.70	94.22
Dataset 3	85.14	14.86	16.52	83.48	84.31
Dataset 4	79.12	20.88	21.56	78.44	78.78

5 Discussion

The features are selected and extracted before the training process in most of the previous approaches. Later, the features are used to train the classification model. The features, which have been used in the previous approaches, are not able to discriminate accurately between the fresh and rotten on this dataset. The reason is the variations of images in this dataset. Here, the convolution layers extract the features for the CNN model to be trained. The proposed CNN model is a series of different layers. The convolutional layers at the beginning of the series learn low-level features. The high-level features are learned by the convolutional layers towards the end of the network. Figure 8 shows the extracted features of the final convolution layer in the proposed CNN model as well as fine-tuned AlexNet for all the datasets. The images in *row 1* show the features extracted by the proposed CNN model and *row 2* shows the features extracted by AlexNet. The features extracted by AlexNet is

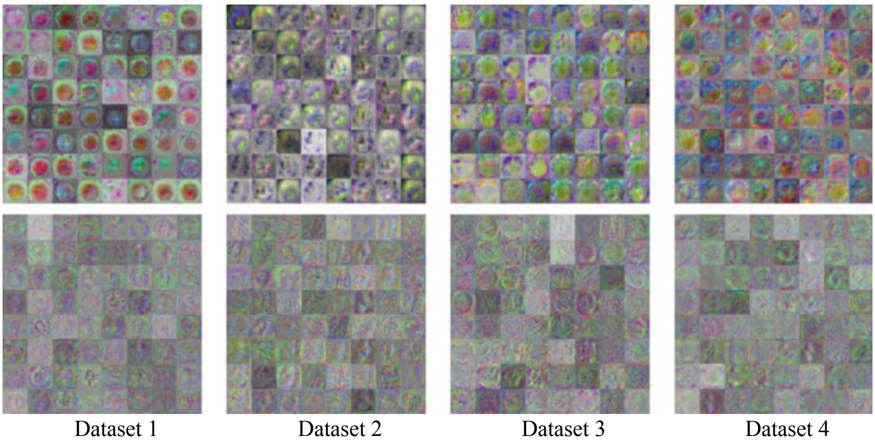


Fig. 8 The extracted features by final convolution layer in (*row* 1) proposed *CNN* model (*row* 2) fine-tuned AlexNet

more complex compared with the proposed CNN model. The reason is that the pre-trained AlexNet has already learned complex feature representation from millions of images of 1000 classes.

The fruits and vegetables are highly perishable. The probability of the rotten sample is less in a lot of recently delivered fruit and vegetable. The number of rotten samples increases in case of transportation delay or storage for too many days. The motivation of this work is to detect and separate rotten fruits and vegetables from a lot of fresh fruits and vegetables as soon as possible. The importance of detecting and sorting the rotten is more than its counterpart. Hence rotten fruits and vegetables are considered as positive classes and fresh fruits and vegetables as negative classes. The samples, which are actually from the positive class and also predicted as the positive class, belongs to True Positive (TP). The samples, which are actually from a positive class and predicted as a negative class, belongs to False Negative (FN). The samples, which are actually from negative class and also predicted as a negative class, belongs to True Negative (TN). The samples, which are actually from a negative class and predicted as a positive class, belongs to False Positive (FP). Refer to Eq. (5) for overall accuracy computation. The overall accuracy using different approaches in the different datasets is shown in Fig. 9. It is visible that the overall classification accuracy using the proposed approach is far better than other approaches for all the datasets.

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{5}$$

The accuracy comparison is not enough for this kind of class imbalance problem. Hence, some specific performance metric is computed for this problem. Recall, precision, and F1 score [40, 41] are computed using Eqs. (6)–(8) correspondingly. The comparison of recall, precision, and F1 score using different approaches in different datasets are shown in Figs. 10, 11, and 12 correspondingly. The figures depict that recall, precision, and F1 score using the proposed approach are comparatively better than prior approaches in all datasets. The performance improvement is upto 50% except dataset 2. The performance improvement is within 20% using the proposed

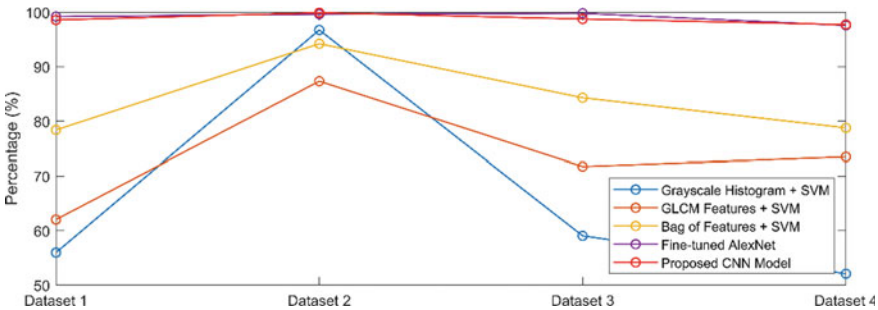


Fig. 9 Comparison of overall accuracy using different approaches in different datasets

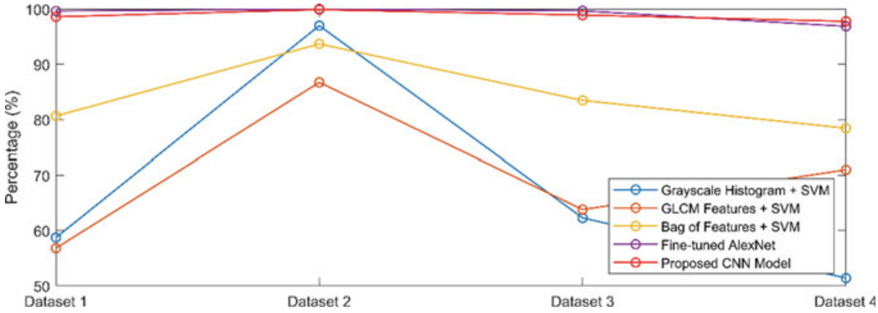


Fig. 10 Comparison of recall using different approaches in different datasets

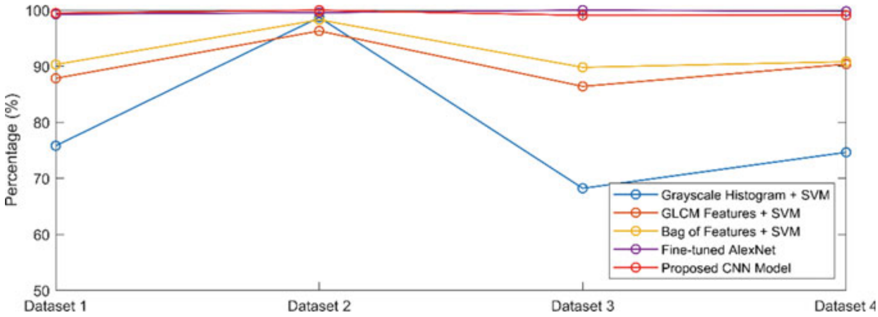


Fig. 11 Comparison of precision using different approaches in different datasets

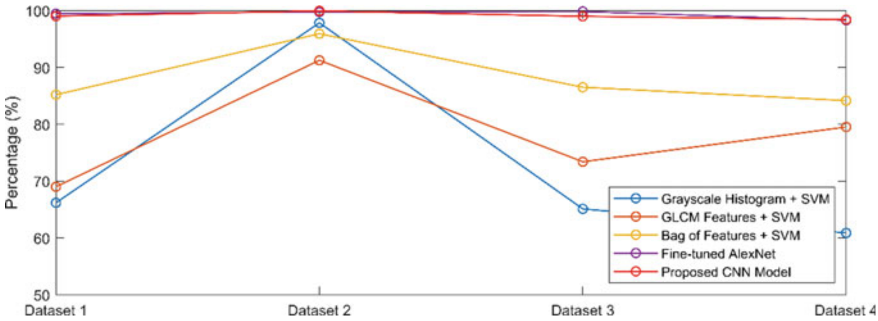


Fig. 12 Comparison of F1 score using different approaches in different datasets

approach for dataset 2, which contains fresh banana and rotten banana. The bag of features model with SVM is the best among the prior techniques in most of the datasets. The difference of F1 score between those two network models is 0.42, 0.18, 0.84, and 0.12% on dataset 1, 2, 3, and 4 correspondingly. The performance wise both the proposed CNN model and fine-tuned AlexNet are very close in the context of this problem. The proposed CNN contains 19 layers whereas the AlexNet contains

25 layers in total. The input size of AlexNet is also more than 3 times of the proposed CNN model. Hence, the training cost is high for AlexNet compared with that of the proposed CNN model.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{7}$$

$$F1\,\text{Score} = \frac{2 * \text{Recall} * \text{Precision}}{\text{Recall} + \text{Precision}} \tag{8}$$

The dataset 1, 2, 3, and 4 consists of three different types of fruit i.e. Apple, Banana, and Orange. Dataset 5 is the collection of 108 images from different sources. Figure 13 shows some samples from dataset 5. It does not include any type of fruit or vegetables from Dataset 1, 2, 3 & 4. Dataset 5 contains 68 fresh fruit and vegetables of 6 different types and 40 rotten fruit and vegetable of 9 different types. The dataset 5 has been created to test the performance of the trained CNN model and AlexNet if the test images come from different sources and of different fruit and vegetable type. The images of dataset 5 fed into the proposed CNN and fine-tuned AlexNet, which are trained with the dataset 4. Table 7 depicts the performance of the proposed CNN

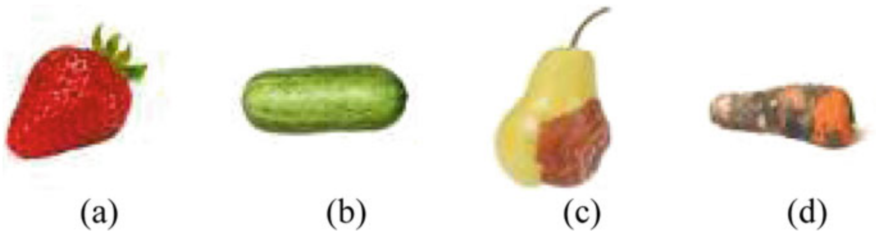


Fig. 13 Samples from Dataset 5 (**a**, **b**) Fresh (**c**, **d**) Rotten

Table 7 The performance using the proposed CNN model and AlexNet on dataset 5

CNN models	Fresh and predicted as fresh (%)	Fresh but predicted as rotten (%)	Rotten but predicted as fresh (%)	Rotten and predicted as rotten (%)	Overall accuracy (%)	Precision (%)	Recall (%)	F1 score (%)
Proposed CNN	95.59	4.41	2.50	97.50	96.30	92.86	97.50	95.12
Fine-tuned AlexNet	95.59	4.41	10.00	90.00	93.52	92.31	90.00	91.14

model and fine-tuned AlexNet on dataset 5. Dataset 4 is a mixture of three types of fruits. Hence, the network models are trained with dataset 4 for testing on dataset 5.

6 Conclusion

The previous approaches mostly proposed for the detection of rotten fruit or vegetable that belongs to a single class e.g. apple, orange, etc. The detection of rotten fruits and vegetables of any class is a limitation for the previous approaches. Convolutional neural network architecture is proposed here to classify fresh and rotten fruits and vegetables of any type. The proposed model has been tested with the fresh and rotten class of three different fruits. It has been also tested with all fresh samples and all rotten samples of three fruits. The proposed model is working very accurately to classify the rotten fruits and vegetables for all sets of data. The performance is stable irrespective of different types of fruit and vegetables. The model is also tested with fruits and vegetable samples of a different type, which are not included in this dataset. The performances on those samples are good as well. The performance of the fine-tuned AlexNet is also as good as the proposed approach. The transfer learning of AlexNet can also be used for the detection of rotten fruits and vegetables. The level of freshness detection can be a very good scope for future research under this problem.

References

1. An, X., Li, Z., Zude-Sasse, M., Tchuenbou-Magaia, F., Yang, Y.: Characterization of textural failure mechanics of strawberry fruit. J. Food Eng. 110016 (2020)
2. Lu, F., Xu, F., Li, Z., Liu, Y., Wang, J., Zhang, L.: Effect of vibration on storage quality and ethylene biosynthesis-related enzyme genes expression in harvested apple fruit. Sci. Hortic. **249**, 1–6 (2019)
3. Singh, D., Sharma, R. R.: Postharvest diseases of fruits and vegetables and their management. In: Postharvest Disinfection of Fruits and Vegetables, pp. 1–52. Academic Press (2018)
4. Cao, J., Wang, C., Xu, S., Chen, Y., Wang, Y., Li, X., Sun, C.: The effects of transportation temperature on the decay rate and quality of postharvest Ponkan (Citrus reticulata Blanco) fruit in different storage periods. Sci. Hortic. **247**, 42–48 (2019)
5. Jidong, L., De-An, Z., Wei, J., Shihong, D.: Recognition of apple fruit in the natural environment. Optik **127**(3), 1354–1362 (2016)
6. Meng, J., Wang, S.: The recognition of overlapping apple fruits based on boundary curvature estimation. In: 2015 Sixth International Conference on Intelligent Systems Design and Engineering Applications (ISDEA), pp. 874–877. IEEE (2015)
7. Xiang, R., Ying, Y., Jiang, H.: Tests of a recognition algorithm for clustered tomatoes based on mathematical morphology. In: 2013 6th International Congress on Image and Signal Processing (CISP), pp. 464–468. IEEE (2013)
8. Xiang, R., Jiang, H., Ying, Y.: Recognition of clustered tomatoes based on binocular stereo vision. Comput. Electron. Agric. **106**, 75–90 (2014)
9. Lv, J., Wang, F., Ma, Z., Rong, H.: Yellow apple recognition method under natural environment. In: 2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics, pp. 46–49. IEEE (2015)

10. Xiang, R., Ying, Y., Jiang, H.: A recognition algorithm for occluded tomatoes based on circle regression. In: 2013 6th International Congress on Image and Signal Processing (CISP), pp. 713–717. IEEE (2013)
11. Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., McCool, C.: Deepfruits: a fruit detection system using deep neural networks. Sensors **16**(8), 1222 (2016)
12. Jana, S., Parekh, R.: Shape-based fruit recognition and classification. In: International Conference on Computational Intelligence, Communications, and Business Analytics, pp. 184–196. Springer, Singapore (2017)
13. Cornejo, J.Y.R., Pedrini, H.: Automatic fruit and vegetable recognition based on CENTRIST and color representation. In: Iberoamerican Congress on Pattern Recognition, pp. 76–83. Springer, Cham (2016)
14. Jana, S., Parekh, R., Sarkar, B.: Automatic classification of fruits and vegetables: a texture-based approach. In: Algorithms in Machine Learning Paradigms, pp. 71–89. Springer, Singapore (2020)
15. Al-falluji, R.A.A.: Color, shape and texture based fruit recognition system. Int. J. Adv. Res. Comput. Eng. Technol. (IJARCET) **5**(7), 2108–2112 (2016)
16. Kuang, H.L., Chan, L.L.H., Yan, H.: Multi-class fruit detection based on multiple color channels. In: 2015 International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR), pp. 1–7. IEEE (2015)
17. Rachmawati, E., Khodra, M.L., Supriana, I.: Histogram based color pattern identification of multiclass fruit using feature selection. In: 2015 International Conference on Electrical Engineering and Informatics (ICEEI), pp. 43–48. IEEE (2015)
18. Zawbaa, H.M., Hazman, M., Abbass, M., Hassanien, A.E.: Automatic fruit classification using random forest algorithm. In: 2014 14th International Conference on Hybrid Intelligent Systems, pp. 164–168. IEEE (2014)
19. Yang, X., Zhang, R., Zhai, Z., Pang, Y., Jin, Z.: Machine learning for cultivar classification of apricots (Prunus armeniaca L.) based on shape features. Sci. Hortic. **256**, 108524 (2019)
20. Zhang, Y., Wang, S., Ji, G., Phillips, P.: Fruit classification using computer vision and feedforward neural network. J. Food Eng. **143**, 167–177 (2014)
21. Hossain, M.S., Al-Hammadi, M., Muhammad, G.: Automatic fruit classification using deep learning for industrial applications. IEEE Trans. Industr. Inf. **15**(2), 1027–1034 (2018)
22. Sa'ad, F.S.A., Ibrahim, M.F., Shakaff, A.M., Zakaria, A., Abdullah, M.Z.: Shape and weight grading of mangoes using visible imaging. Comput. Electron. Agric. **115**, 51–56 (2015)
23. Nandi, C.S., Tudu, B., Koley, C.: A machine vision technique for grading of harvested mangoes based on maturity and quality. IEEE Sens. J. **16**(16), 6387–6396 (2016)
24. Jana, S., Parekh, R., Sarkar, B.: Volume estimation of non-axisymmetric fruits and vegetables using image analysis. In: 2019 International Conference on Computing, Power and Communication Technologies (GUCON), pp. 628–633. IEEE (2019)
25. Gokul, P. R., Raj, S., Suriyamoorthi, P.: Estimation of volume and maturity of sweet lime fruit using image processing algorithm. In: 2015 International Conference on Communications and Signal Processing (ICCSP), pp. 1227–1229. IEEE (2015)
26. Iqbal, S.M., Gopal, A., Sarma, A.S.V.: Volume estimation of apple fruits using image processing. In: 2011 International Conference on Image Information Processing, pp. 1–6. IEEE (2011)
27. Vivek Venkatesh, G., Iqbal, S.M., Gopal, A., Ganesan, D.: Estimation of volume and mass of axi-symmetric fruits using image processing technique. Int. J. Food Prop. **18**(3), 608–626 (2015)
28. Chandini, A.A., Maheswari B., U.: Improved Quality Detection Technique for Fruits Using GLCM and MultiClass SVM. In: 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 150–155. IEEE (2018)
29. Karakaya, D., Ulucan, O., Turkan, M.: A comparative analysis on fruit freshness classification. In: 2019 Innovations in Intelligent Systems and Applications Conference (ASYU), pp. 1–4. IEEE (2019)

30. Yogesh, Dubey, A.K., Ratan, R., Rocha, A.: Computer vision based analysis and detection of defects in fruits causes due to nutrients deficiency. Cluster Comput. 1–10 (2019)
31. Gómez-Sanchis, J., Martín-Guerrero, J.D., Soria-Olivas, E., Martínez-Sober, M., Magdalena-Benedito, R., Blasco, J.: Detecting rottenness caused by Penicillium genus fungi in citrus fruits using machine learning techniques. Expert Syst. Appl. **39**(1), 780–785 (2012)
32. Kamalakannan, A., Rajamanickam, G.: Surface defect detection and classification in mandarin fruits using fuzzy image thresholding, binary wavelet transform and linear classifier model. In: 2012 Fourth International Conference on Advanced Computing (ICoAC), pp. 1–6. IEEE (2012)
33. Capizzi, G., Sciuto, G.L., Napoli, C., Tramontana, E., Woźniak, M.: Automatic classification of fruit defects based on co-occurrence matrix and neural networks. In: 2015 Federated Conference on Computer Science and Information Systems (FedCSIS), pp. 861–867. IEEE (2015)
34. Ranjit, K.N., Naveen, C., Chethan, H.K.: Fruit disease categorization based on color, texture and shape features. Int. J. Comput. Appl. **178**(49), 16–19 (2019)
35. Jha, S.N., Matsuoka, T.: Development of freshness index of eggplant. Appl. Eng. Agric. **18**(5), 555 (2002)
36. Roy, K., Chaudhuri, S.S., Bhattacharjee, S., Manna, S., Chakraborty, T.: Segmentation techniques for rotten fruit detection. In: 2019 International Conference on Opto-Electronics and Applied Optics (Optronix), pp. 1–4. IEEE (2019)
37. Roy, K., Ghosh, A., Saha, D., Chatterjee, J., Sarkar, S., Chaudhuri, S.S.: Masking based segmentation of rotten fruits. In: 2019 International Conference on Opto-Electronics and Applied Optics (Optronix), pp. 1–4. IEEE (2019)
38. Kalluri, S. R.: Fruits fresh and rotten for classification. https://www.kaggle.com/sriramr/fruits-fresh-and-rotten-for-classification, last accessed 14 Feb 2020
39. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
40. Karlekar, A., Seal, A.: SoyNet: Soybean leaf disease classification. Comput. Electron. Agric. **172**, 105342 (2020)
41. Al Imran, A., Rifat, M.R.I., Mohammad, R.: Enhancing the classification performance of lower back pain symptoms using genetic algorithm-based feature selection. In: Proceedings of International Joint Conference on Computational Intelligence, pp. 455–469. Springer, Singapore (2020)

Deep Learning-Based Essential Paddy Pests' Filtration Technique for Economic Damage Management

Md. Zahid Hasan, Nahid Zeba, Sumaita Binte Shorif, and Morium Akter

1 Introduction

Bangladesh is developing as an agricultural country with 14.3 million Ha of total agricultural land where almost 59.8% of the land can be used for cultivation [1]. Almost 85% of the total people are directly or indirectly dependent on agriculture for their living [2]. The researcher found that paddy took around 76% of the total cropped area which provides 95% of cereal foods for people [3]. Due to several attacks by pests in paddy plants, a large amount of potential yield is wasted [4]. Compared to other developed countries, most of our farmers are illiterate and do not have the least knowledge about farming and agriculture. For lack of adequate knowledge, farmers can not differentiate between beneficial and harmful pests; therefore, whenever they see any kind of pest, they tend to spray insecticides killing useful pests of paddy and the chemicals of insecticides cause damages of crops which decreases productivity. Previously researchers mainly focused on harmful pest detection of paddy plants.

This chapter emphasizes the automated pest detection and identification system which can also be called an agro-medical expert system for paddy using computer vision and deep learning technique (see Fig. 1). It is an effective mobile supportive

Md. Z. Hasan (✉) · N. Zeba
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh
e-mail: zahid.cse@diu.edu.bd

N. Zeba
e-mail: nahid25-863@diu.edu.bd

S. B. Shorif · M. Akter
Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh
e-mail: sumaita.bs@gmail.com

M. Akter
e-mail: morium.akter@juniv.edu

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_4

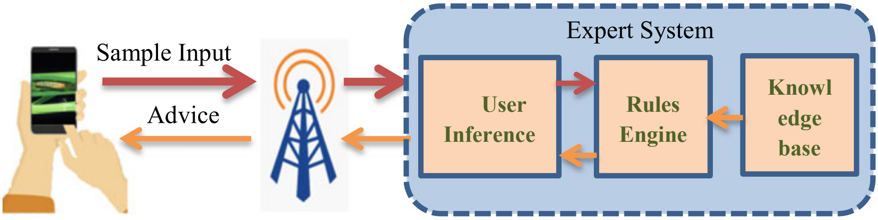


Fig. 1 Expert system of detecting and identifying beneficial and non-beneficial pests

expert system for our farmers that can detect both the beneficial and damaging pests for better productivity of their crops. If we can increase the productivity and quality of our crops, it will enrich the economy of Bangladesh and will resolve the poverty issues.

Deep learning is showing a significant impact on vision-based detection and identification system. It is a subfield of machine learning which concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. Recently, most of the countries are using deep learning techniques for analyzing different kinds of diseases of tomato [5], peach [6], tea [7], and apple [8] just from the image of their leaves. Computer vision extracts the information from the image automatically, and deep learning analyzes that information for giving accurate output [9]. The author used convolutional neural networks (CNNs) method for image processing which comes under artificial neural networks (ANNs) that uses perceptron, a machine learning algorithm for supervised learning to analyze data. Within a very short time, CNN can process a large amount of data. When it comes to the CNN method, the level of pest detection and identification is very high and accurate. A massive amount of data, sufficient training, and computational resources are needed to train the system for deep learning so that it can give accurate output. For training, the image should be collected from real-life environments and pictures must be taken from different angles for better analyzing. The authors try to introduce this system to the farmers so that they can use it very easily by taking pictures of the crops on their mobile devices enabling the system to detect and identify the pest automatically by using the algorithm. Based on the results, the farmer can then decide if he needs to spray any medicine on the crops or not.

In short, the objective of the project is to build a system that can differentiate between the harmful and beneficial pests of paddy from real-life environmental images using the CNN method of deep learning techniques which is easily supported by mobile devices. In that case, it can be a very effective pest detecting and identification tool for farmers.

The remaining sections of this chapter are organized as follows. Section 2 describes related literature. Section 3 introduces beneficial and non-beneficial paddy pests. Section 4 presents the methodology framework for this research. Section 5 illustrates our investigated deep learning strategies. Section 6 shows our experimental results along with datasets. Finally, Sect. 7 concludes this chapter.

2 Related Works

In recent years, automated pest detection and identification of crops have become very active research topics. Most of the case researchers focused only on the destructive pest of crops. On the contrary, this chapter is focusing on identifying and detecting both destructive and useful pests of paddy plants so that farmers can decide when to use insecticides to reduce a large number of damages. About 90% of the farmers of Bangladesh use pesticides in their crops unnecessarily at least once [10].

Computer vision and deep learning show a great impact on object detection and identification. Previously image extraction and classification works were done by feature-based detection algorithms like SURF and SIFT [11] which was considered a very hard process; also after changing the dataset, it was needed to be revisited. But the development of computer vision and deep learning upgrades the situation extremely and improve the level of accuracy for detecting and identifying harmful/useful pests from real-life environmental pictures.

Gutierrez et al. [12] worked on the detection and identification of some destructive pests using their captured data with the help of K-nearest neighbor (KNN) and multilayer perceptron (MLP). KNN gives 66.47% accuracy, and MLP gives 81.12% accuracy.

Every year pecan weevil caused huge damages in the paddy field which became a matter of concern for farmers. Al-Saqer [13] tried to focus on that fact, and for solving the problem, he proposed a model that can detect harmful weevil pests. For detecting the pecan weevil in the paddy field, he developed a neural network-based identification system using captured images. But his focus was only on pest pecan weevil. There is another research team [14] who were mainly concerned about tomato plant diseases attacked by only borers. They proposed a very simple algorithm for processing the image that can detect pests of tomato plants at an early age. The limitation of that work is just focusing on only the borer pests.

In China, maize is considered one of the major foods of the country. As cultivation land started decreasing for urbanization, the quality of maize seeds became a growing concern for the nation. Reference [15] focused only on identifying the defect of maize seeds by using two methods GoogleNet and Cifario. Around 500 images were taken for experimentation, and the obtained accuracies were 98.9% (GoogleNet) and 98.8% (Cifario). Sushmitha et al. [16] surveyed destructive pest and disease identification of crops using different deep learning methods to find the accuracy level of the specific architecture. Their main focus was on measuring the accuracy of different methods rather than pest detection. From the year 1998 to 2017, different types of CNN architectures were used for pest and disease detection and identification.

From the above literature survey, all the researchers detected and identified different destructive pests on different plants but none of them talked about beneficial pests of the paddy. Besides, some works used a few images for evaluating which may cause data biasing and become less efficient in the practical field. By identifying beneficial pests of the rice plant, unnecessary use of pesticides and productivity damage can be prevented.

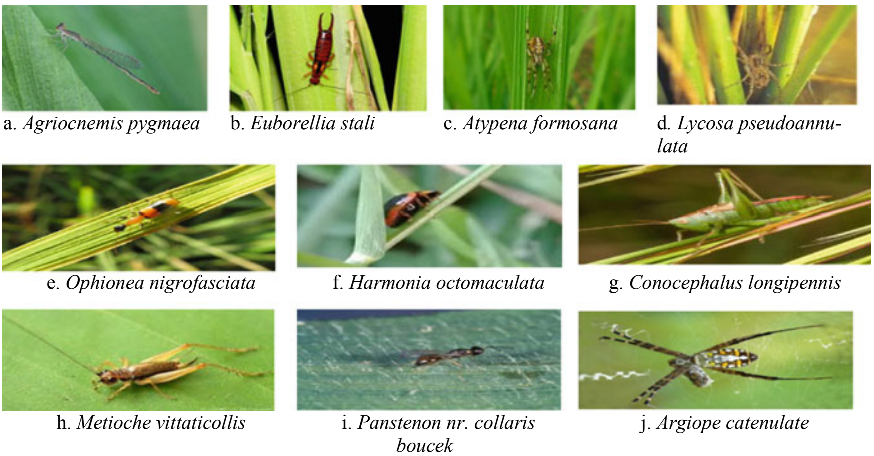
3 Pests Classification

In Bangladesh, paddy plants can be attacked by more than 159 species of pests, and among them, 20–23 species can cause huge economical damage every year [17]. There can be two kinds of pests in crops, some pests are beneficial and some are non-beneficial or harmful.

3.1 Beneficial Pests

Some pests are beneficial for paddy plants as they eat and kill destructive pests like leafhopper and planthopper, and some play an important role in controlling the ecosystem of the rice field. They are also called healthy pests of paddy. Damselflies (*Agriocnemis pygmaea*), earwig (*Euborellia stali*), field spider (*Atypena formosana*), wolf spider (*Lycosa pseudoannulata*), ground beetles (*Ophionea nigrofasciata*), large spotted ladybird (*Harmonia octomaculata*), meadow grasshopper (*Conocephalus longipennis*), sword-tailed cricket (*Metioche vittaticollis*), wasp (*Panstenon nr—collaris* boucek), web spinning orb-weaver spider (*Argiope catenulata*) are beneficial pests of rice crops (see Fig. 2).

Agriocnemis pygmaea or Damselflies are some of the important predators of paddy pests. Lady beetle catches mainly different harmful pests like planthoppers and leafhoppers. The ground beetle is considered a generalist predator. Sword-tailed cricket is the enemy of leafhoppers, planthoppers, eggs of armyworms, and stem borers.



a. *Agriocnemis pygmaea* b. *Euborellia stali* c. *Atypena formosana* d. *Lycosa pseudoannulata*

e. *Ophionea nigrofasciata* f. *Harmonia octomaculata* g. *Conocephalus longipennis*

h. *Metioche vittaticollis* i. *Panstenon nr. collaris boucek* j. *Argiope catenulate*

Fig. 2 Beneficial pests of paddy field

3.2 Non-beneficial Pests

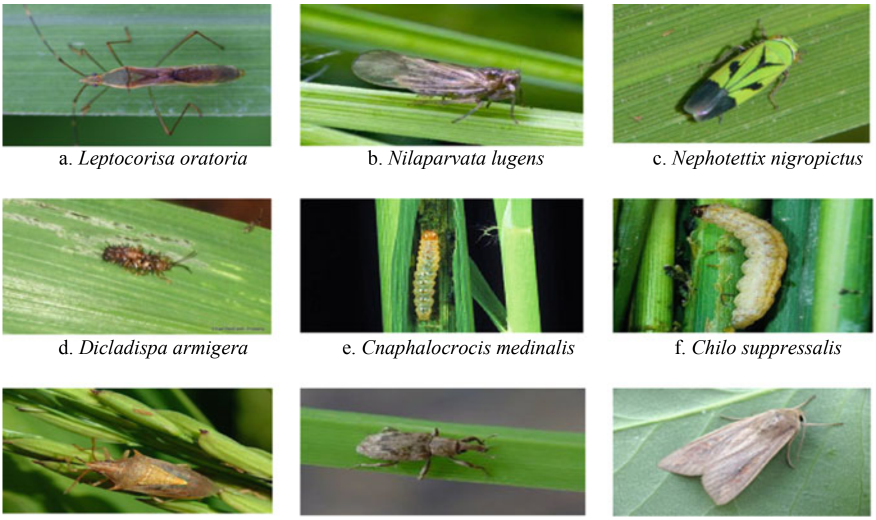
Some destructive or non-beneficial pests of paddy plants are Gandhi bug/rice Bug (*Leptocorisa oratoria*), brown planthopper (*Nilaparvata lugens*), rice green leafhopper (*Nephotettix nigropictus*), rice hispa (*Dicladispa armigera*), rice leaf roller (*Cnaphalocrocis medinalis*), rice stem borer (*Chilo suppressalis*), stink bug (*Oebalus pugnax*), rice water weevil (*Lissorhoptrus oryzophilus*), rice armyworm moth (*Mythimna unipuncta*), etc. (see Fig. 3).

Rice stem borer is considered as the most destructive pest because it damages the paddy plants from stem to adult level. Deadhead and whitehead are two types of damages caused by Stem borer [18]. Rice leafhopper and Planthopper are responsible for the complete drying of paddy plants. Paddy plants are eaten by stink bug which causes huge damage. Rice grain weevil not only attack the crops but also eat them.

4 Methodology

This study aims to make a model based on deep convolutional neural networks to identify pests from user-provided images (see Fig. 4). Figure 4 explains a flowchart of the identification process of beneficial and not beneficial paddy pests with a name. All blocks in this diagram will be discussed in the following sections.

The main focus of this chapter is detecting and identifying beneficial and non-beneficial pests with computer vision and deep learning techniques. As detecting



a. *Leptocorisa oratoria* b. *Nilaparvata lugens* c. *Nephotettix nigropictus*

d. *Dicladispa armigera* e. *Cnaphalocrocis medinalis* f. *Chilo suppressalis*

g. *Oebalus pugnax* h. *Lissorhoptrus oryzophilus* i. *Mythimna unipuncta*

Fig. 3 Destructive pests of paddy field

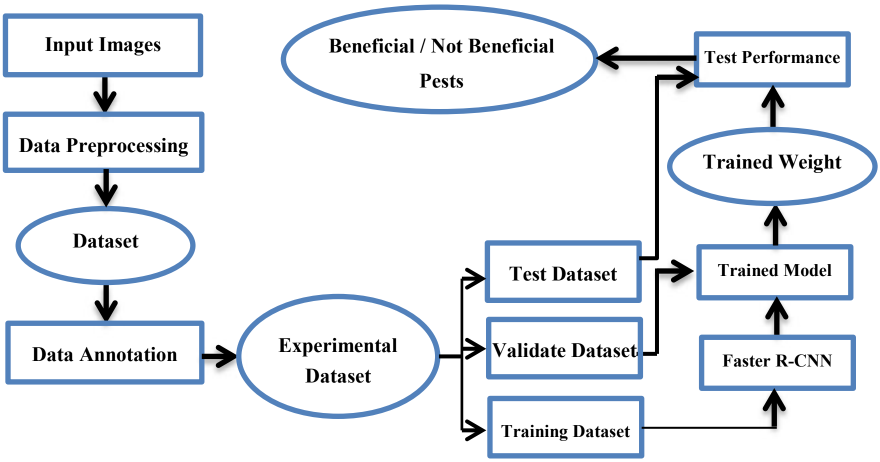


Fig. 4 Flowchart for pest's identification of paddy

and identifying techniques are based on learning algorithms, it is very important to generate and label the dataset of different types of beneficial and non-beneficial pests that were generated manually and automatically from the set of images. Dataset quality and its labeling will greatly impact on the accuracy of the system. For that reason, generated pictures are labeled using image annotation tools for getting accurate results of beneficial and non-beneficial pests. We have done some different image pre-processing steps such as denoising [19, 20], contrast enhancement [21], and background subtraction [22].

5 Deep Learning

Deep learning is a field of machine learning concerned with algorithms that are inspired and structured by the function of the brain called artificial neural networks.

5.1 *Convolutional Neural Network*

A powerful technology called convolutional neural network (CNN) is used for the classification of visual inputs that are raised from documents [23]. It is found [24] that a form of transitional invariance is acquired when neurons with the same parameters are applied on previous layer patches at different locations.

CNN consists of three types of neural layers, namely convolutional layers, pooling layers, and fully connected layers which is responsible for performing different kinds of operations. For detecting and identifying beneficial and non-beneficial pests from the image the author used CNN methods.

5.1.1 Convolutional Layers

This layer receives the input and transforms the input and output into the next layer which operation is called convolutional operations. It can detect the pattern of the images using filters. The image input needs to be given, and the CNN utilizes various kernels to convolve the whole image along with intermediate feature maps, generating different types of feature maps.

5.1.2 Pooling Layers

The function of the pooling layer is to progressively reduce the structural size of the representation to reduce the number of parameters and computation in the network. In this layer, image is shrinking into smaller sizes or reducing the size of each image that causes simultaneous loss of information which is called subsampling or downsampling [25]. The depth dimension (width × height) of the volume is not affected by the layer. The decreasing size of the image is considered beneficial because lower in size leads to less computational over the head for the upcoming layers which works against overfitting.

5.1.3 Fully Connected Layers

The fully connected layer is working as feed-forward neural networks. This is the final layer of CNN where actual classification happens. Featured maps are converted into featured vectors in a fully connected layer which we get from previous layers. To implement forward and backward propagation fully, the connected layer follows some rules [23] given below:

$$x_j^{L+1} = \sum_i w_{j,i}^{L+1} x_i^L \tag{1}$$

$$g_j^L = \sum_j w_{j,i}^{L+1} g_j^{L+1} \tag{2}$$

Here, x_i^L and g_i^L are, respectively, the activation and gradient unit of i at layer L, and $w_{j,i}^{L+1}$ is weight connecting unit i at layer L to unit j at layer $L + 1$.

Neurons are capable of extracting features such as corners and edges and combined with the subsequent convolutional layer for detecting high order features [25]. Each plane is responsible for constructing feature maps which is equivalent to convolutional operations followed by additive bias term and sigmoid function:

$$y^{(d)} = \sigma(\mathbf{w}y^{(d-1)}) + \mathbf{b} \tag{3}$$

Here, d is the depth of the convolutional layer, w is the weight matrix, and b is bias term. The weight matrix is whole for fully connected neural networks that connect with all the inputs to every unit with different weights.

5.1.4 Faster R-CNN

The methods, structures, and detection process which are used in this chapter are described in this section. The Faster R-CNN is an object detection architecture that is used to identify and detect the beneficial and non-beneficial pests of paddy from image (see Fig. 5). To obtain high recognition rates of beneficial and non-beneficial paddy pest features extraction, neural networks must accurately extract characteristics of the images containing paddy pests.

ResNet-101, VGG-16, and MobileNet are three types of deep convolutional neural networks which used to extract image features in this chapter. In Faster R-CNN, from input images, deep convolutional neural network (DCNN) extracts feature maps and region proposals that are generated through region proposal network (RPN) and fully connected layer. After that fixed-sized features, map generated for classification and positioning.

For each region of interest (ROI), there are two outputs of Faster R-CNN, one is a classification result which labels the boxes and another is a regression result that provides the coordinates of region proposals. On the other hand, the final input of a fully connected layer needs consistency in size so ROIs which are consistent in size are used as input in a fully connected layer.

5.1.5 VGG-16

In the VGG-16 network, the first and second convolutional layers contain 64 feature kernel filters where the filter size is 3×3. RGB image with depth 3 passes through those layers and dimension changes into $224 \times 224 \times 64$, and the output is passed to the max-pooling layer with a stride of 2 [26].

The third and fourth layers have 124 feature kernel filters with 3×3 size filter, and these are followed by a max-pooling layer with stride 2 where output is reduced

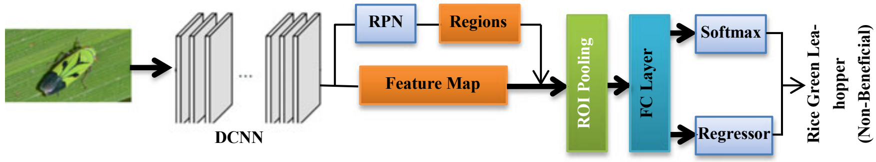


Fig. 5 Architecture of Faster R-CNN

to 56 × 56 × 128. The fifth, sixth, and seventh layers use 256 feature maps with stride 2, and the eighth to thirteen have 512 kernel filters with stride 1. Fourteen and fifteen layers are fully connected hidden layers of 4096 units followed by 1000 units of the softmax output layer [26].

5.1.6 ResNet-101

In this chapter, ResNet-101 is a convolutional neural network of 101 layers deep with three layers of the block that is used for extracting images as it gives more accurate results than any other layers by considerable margins [27]. For scale augmentation, the picture needs to be reshaped with its shorter side and 224 × 224 crop which is sampled from the picture with per-pixel mean subtracted. The total duration of the process was 22.50 h with a large number of iterations. The significant accuracy is achieved from the considerably increased depth which is witnessed for the evaluation matrices.

5.1.7 MobileNet

MobileNet is nearly as accurate as VGG-16 which is 32 times smaller, 27 times less compute-intensive, and more accurate than any other networks [28]. It is the most effective base network in the modern pest detection system. The author trained MobileNet for recognition of the beneficial and non-beneficial paddy pests' dataset which is larger and noisy, and results achieve the desired state followed by greatly reduced computation and size.

6 Experiments

The details of the experiments, for example, data collection, parameter fitting, and experimental results all are discussed in this section of the chapter. Every model's performance which is used during the whole process is shown through data and images.

6.1 Dataset

We have acquired 673 images of beneficial pests and 805 images of non-beneficial pests. These images are expanded through augmentation (by scaling and rotations) to 3365 images for the beneficial pests and 6930 images for the non-beneficial pests. The whole experimentation is performed at a ratio of 6:2:2 for training, validation, and testing, respectively.

Table 1 Categorical information on the beneficial pest dataset

Beneficial pest	Original[a]	Expanded[b]	Training[c]	Validation[d]	Test[e]
Damselflies	54	270	162	54	54
Earwig	66	330	198	66	66
Field spider	102	510	306	102	102
Wolf spider	72	360	216	72	72
Ground beetles	42	210	126	42	42
Large spotted ladybird	64	320	192	64	64
Meadow grasshopper	113	565	339	113	113
Sword-tailed cricket	84	420	252	84	84
Wasp	32	160	96	32	32
Web spinning orb-weaver spider	44	220	132	44	44
Total	673	3365	2019	673	673

[a]Number of original images
[b]Number of images after augmentation
[c]Number of trained set images
[d]Number of images in the validation set
[e]Number of images in the test set

The categorical information on beneficial and non-beneficial pest datasets is shown in Tables 1 and 2, respectively.

6.2 Experimental Setup

To train the model, mainly training set is used and the validation set is used for giving feedback about the progress of the training and regulates the training if it is complete or not. Lastly, the trained model is applied to the test set for evaluating the model's performance.

In the experiment, a very small dataset is used and the number of iterations of the Faster R-CNN was set to 70,500 where the training results are saved every 5540 iterations.

The three different models ResNet-101, VGG-16, and MobileNet are used.

6.3 Confusion Matrix

A confusion matrix is a size of $n \times n$ with a classifier that shows the predicted and actual classification where n is the number of different classes [29]. In the confusion matrix, TP, TN, FP, and FN indicate true positive, true negative, false positive, and

Table 2 Categorical information on the non-beneficial pest dataset

Non-beneficial pest	Original[a]	Expanded[b]	Training[c]	Validation[d]	Test[e]
Gandhi bug/rice bug	44	220	132	44	44
Brown pant hopper	67	335	201	67	67
Rice green leafhopper	72	360	216	72	72
Rice hispa	50	250	150	50	50
Rice leaf roller	109	545	327	109	109
Rice stem borer	39	195	117	39	39
Stink bug	47	235	141	47	47
Rice water weevil	78	390	234	78	78
Rice armyworm moth	101	505	303	101	101
Gandhi bug/rice bug	86	430	258	86	86
Brown planthopper	112	3465	2079	693	693
Total	805	6930	4158	1386	1386

[a]Number of original images
[b]Number of images after augmentation
[c]Number of trained set images
[d]Number of images in the validation set
[e]Number of images in the test set

false negative, respectively. Accuracy is a ratio of correctly predicted observations to the total observations, and it is calculated as,

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{FN} + \text{TN}} \tag{4}$$

Beneficial and non-beneficial pest's images, accuracy of ResNet-101 is highest 96.93% and 95.07%, respectively (see Figs. 6 and 7).

Fig. 6 Beneficial pest detection accuracy

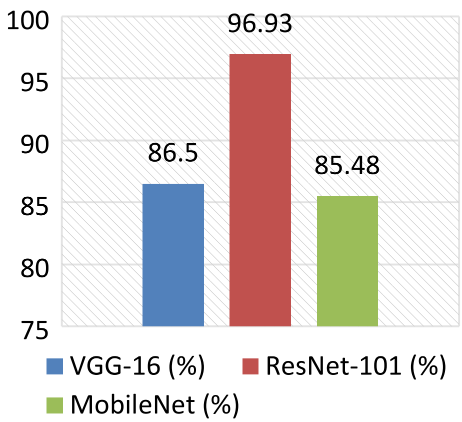
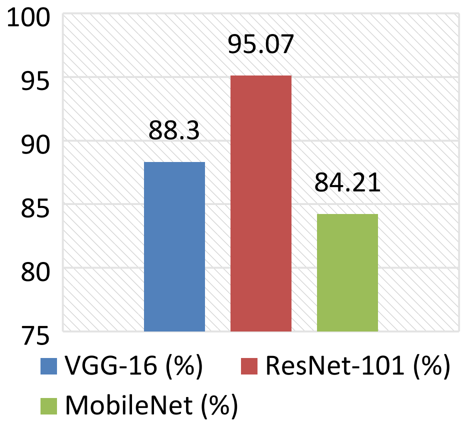


Fig. 7 Non-beneficial pest detection accuracy



The VGG-16 gets the highest value in two types of pest detection, while in Table 3, ResNet-101 performs well to detect nine types of beneficial pest, and in Table 4, it detects eight types of non-beneficial pests from image (see Tables 3 and 4).

Table 3 Obtained accuracy by using different CNN models for the beneficial pests' dataset

Beneficial pest	VGG-16 (accuracy in %)	ResNet-101 (accuracy in %)	MobileNet (accuracy in %)
Damselflies	97.99	97.99	97.98
Earwig	79.24	96	67
Field spider	90.43	97	71.7
Wolf spider	79.8	97	97.5
Ground beetles	98	93.33	90.39
Large spotted ladybird	94.18	98	96
Meadow grasshopper	90.62	98	97.97
Sword-tailed cricket	94.2	96	68.26
Wasp	70	98	77.44
Web spinning orb-weaver spider	70.59	98	90.54
Accuracy	86.50	96.93	85.48

Table 4 Obtained accuracy by using different CNN models for the non-beneficial pests' dataset

Non-beneficial pest	VGG-16 (accuracy in %)	ResNet-101 (accuracy in %)	MobileNet (accuracy in %)
Gandhi bug/rice bug	94.18	98.56	97.48
Brown planthopper	98.99	98.75	98
Rice green leafhopper	90.2	94	93
Rice hispa	79.24	96	67
Rice leaf roller	77.9	96	66.26
Rice stem borer	99	99.43	93
Stink bug	70.59	85	70.7
Rice water weevil	94.2	96	95
Rice armyworm moth	90.43	91.86	77.44
Accuracy	88.30	95.07	84.21

6.4 Computation Time

Three different neural networks were used to detect pests from the image and showed that those proposed methods give very accurate results in detecting and identifying beneficial and non-beneficial pests. Results of the experiment indicate that ResNet-101 gives that highest detection rate but for training and detecting pests, it takes more time than the other two methods (see Table 5). On the other hand, MobileNet gives less accurate results but for training purposes, it takes less time than ResNet-101.

Based on the above table results, it can be said that the deeper and well-performed convolutional neural network has a more complex structure which takes a long time to detect objects than the normal methods.

7 Conclusion

In this chapter, we focused on the identification and detection of beneficial and harmful pests of paddy field using a region-based Faster R-CNN, where a deep CNN is connected with a region proposal network (RPN). It is found that ResNet-101 gives more accurate results of detecting pest from an image than VGG-16 and MobileNet but takes a long time due to its complex structure. ResNet-101 got the

Table 5 Training time of three models

Faster R-CNN	Models	Time (h)
	ResNet-101	22.50
	MobileNet	13.42
	VGG-16	19.25

highest accuracy of 96.93% and 95.07% in detecting beneficial and non-beneficial pests, respectively. Through this system, a farmer can identify the pests in a real field by taking images by a smartphone. This will help the farmer to take the required action. Future work will be focused on finding a faster computational deep learning model for learning with a better accuracy level.

References

1. Agriculture, https://en.banglapedia.org/index.php?title=Agriculture#:~:text=Agricultural%20land%20The%20total%20land,and%20very%20lowland%20(2%25), last accessed 16 July 2020
2. Chowdhury, A.T., Chowdhury, S.S.: Performance Evaluation of Agricultural Banks in Bangladesh. Int. J. Bus. Manage. **6**(4), 75–89 (2011)
3. Ahiduzzaman, M.: Rice husk energy technologies in Bangladesh. Agric. Eng. Int. CIGR Ej. **9** (2007)
4. Zadoks, C. J.: Crop loss assessment: a historical perspective and rationale. In: International Workshop on Crop Loss Assessment to Improve Pest Management in Rice and Rice-based Cropping Systems in South and Southeast Asia, pp. 1–9. International Rice Research Institute, Philippines (1990)
5. Fuentes, A., Yoon, S., Kim, S.C., Park, D.S.: A robust deep-learning-based detector for real-time tomato plant diseases and pest's recognition. J. Sens. **17**(9) (2017)
6. Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D., Stefanovic, D.: Deep neural networks-based recognition of plant diseases by leaf image classification. Comput. Intell. Neurosci. (2016)
7. Karmokar, B.C., Ullah, M.S., Siddiquee, M.K., Alam, K. M. R.: Tea leaf diseases recognition using neural network ensemble. Int. J. Comput. Appl. **114**(17) (2015)
8. Wang, G., Sun, Y., Wang, J.: Automatic image-based plant disease severity estimation using deep learning. Comput. Intell. Neurosci. (2017)
9. Huang, S., Fan, X., Sun, L., Shen, Y., Suo, X.: Research on classification method of maize seed defect based on machine vision. J. Sens. 1–9. Hindawi, London (2019)
10. Miah, S., Hoque, A., Paul, D., Rahman, D.: Unsafe use of pesticide and its impact on health of farmers: a case study in Burichong Upazila, Bangladesh. IOSR J. Environ. Sci. Toxicol. Food Technol. **8**(1), 57–67 (2014)
11. Salahat, E., Qasaimeh, M.: Recent advances in features extraction and description algorithms: a comprehensive survey. In: Proceedings of the IEEE International Conference on Industrial Technology (ICIT), pp. 1059–1063 (2017)
12. Gutierrez, A., Ansuategi, A., Susperregi, L., Tubío, C., Rankić, I., Lenža, L.: A benchmarking of learning strategies for pest detection and identification on tomato plants for autonomous scouting robots using internal databases. J. Sens. 1–15 (2019)
13. Al-Saqer, S.M.: A robust recognition system for pecan weevil using artificial neural networks. Am. J. App. Sci. (2012)
14. Prathibha, G.P., Goutham, T., Tejaswini, M., Rajas, P.R., Balasubramani, K.: Early pest detection in tomato plantation using image processing. Int. J. Comput. Appl. **96**(12), 22–24 (2014)
15. Zhang, X., Qiao, Y., Meng, F., Fan, C., Zhang, M.: Identification of Maize leaf diseases using improved deep convolutional neural networks. IEEE Access **6**, 30370–30377 (2018)
16. Aryalekshmi, B.N., Rajashekar, B., Sushmitha, M.T.: Survey on deep learning architectures in identification of crop pests and diseases. Int. J. Adv. Sci. Technol. 29(10S), 8274–8281 (2020)
17. Nasiruddin, M., Roy, R.: Rice field insect pests during the RRCE growing seasons in two areas of Hathazari, Chittagong. Bangladesh J. Zool. **40**(1), 89–100 (2012)

18. Pathak, M., Khan, Z.: Insect Pests OF RICE. International Rice Research Institute, Philippines (1994)
19. Brummer, B., Vleeschouwer, D.C.: Natural Image Noise Dataset. IEEE Xplore, Louvain-la-Neuve, Belgium (2020)
20. Deng, G., Cahill, L, W. An adaptive Gaussian filter for noise reduction and edge detection. In: Proceedings of Nuclear Science Symposium and Medical Imaging Conference, (1993)
21. Firdausy, K., Sutikno, T., Prasetyo, E.: Image enhancement using contrast stretching on RGB and IHS digital image. Telkomnika (Telecommunication Computing Electronics and Control), vol. 5, pp. 45, (2007)
22. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: principles and practice of background maintenance. In: Proceedings of the Seventh IEEE International Conference on Computer Vision 1999, vol. 1, pp. 255–261. Greece (1999)
23. Simard, P., Steinkraus, D., Platt, J.: Best practices for convolutional neural networks applied to visual document analysis. In: Seventh International Conference on Document Analysis and Recognition, Proceedings (2003)
24. Fukushima, K.: Neocognitron: a self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. Biol. Cybern. **36**(4), 193–202 (1980)
25. Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E.: Deep learning for computer vision: a brief review. Comput. Intell. Neurosci. **2018**, 1–13 (2018)
26. Tammina, S.: Transfer learning using VGG-16 with deep convolutional neural network for classifying images. Int. J. Sci. Res. Publ. (IJSRP) **9**(10) (2019)
27. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)
28. Howard, A., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Adam, H.: MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications, https://arxiv.org/abs/1704.04861, last accessed 19 July 2020
29. Hasan, M.Z., Hasan, K.M.Z., Sattar, A.: burst header packet flood detection in optical burst switching network using deep learning model. Proc. Comput. Sci. **14**, 970–977 (2018)

Deep CNN-Based Mango Insect Classification

Muntaka Rahat, Mehedi Hasan, Md. Mahmodul Hasan, Md. Tarequl Islam, Md. Syfur Rahman, A. K. M. Kamrul Islam, and Mohammad Motiur Rahman

1 Introduction

Mango insects are nowadays a big threat to the cultivation of mango. Attack of mango insects can have a severe outcome on the quality and quantity of mango production. A study on the impact of fruit fly on mango production indicated that production decreases from 24% to a higher loss of 60% [1]. If the insects can be classified adequately, then the cultivation of mango will give a significant improvement in

M. Rahat · M. Hasan · Md. M. Hasan · A. K. M. K. Islam · M. M. Rahman (✉)
Department of Computer Science and Engineering, Mawlana Bhashani Science and Technology University, Santosh, Tangail 1902, Bangladesh
e-mail: mm73rahman@gmail.com

M. Rahat
e-mail: muntakace15036@gmail.com

M. Hasan
e-mail: mehedicse15030@gmail.com

Md. M. Hasan
e-mail: mahmodul.mbstu@gmail.com

A. K. M. K. Islam
e-mail: kamrul3000@gmail.com

Md. T. Islam
Department of Information and Communication Technology, ICT Division, Dhaka, Bangladesh
e-mail: tareq07@gmail.com

Md. S. Rahman
Education Management Information System (EMIS) Cell, Ministry of Education, Dhaka, Bangladesh
e-mail: syfur022@gmail.com

A. K. M. K. Islam
Department of Computer Science, Georgia State University, Atlanta, GA 30302-5060, USA

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_5

production. Almost a dozen insects have been found damaging the mangoes, which cause severe economic losses. The fruit fly, mango hopper, mango stone weevil, mealybug, stem borer, leaf weevil, red ant, and others are termed as the significant mango insects [2]. Distinguishing the proper category of insects is essential to take necessary actions. The diversity of these insects makes it rigid to classify a particular category correctly. However, a computer-aided solution can capture the diversity of insects to find a solution. This research proposes a scheme to classify mango insects.

In recent years, tremendous research has been conducted on insect classification based on advanced techniques such as machine learning or deep learning. However, there exist not many thorough studies on mango insect classification due to the unavailability of the dataset. Machine learning-based classification techniques require sophisticated datasets to perform analysis. However, most classification methods for image data use deep learning algorithms rather than traditional machine learning methods. The advances in computation resources make it feasible to use deep learning techniques instead of others. In this study, we also proposed a deep learning method to detect mango insects. This research experimented with different architectures of convolutional neural networks (CNN). Besides, we studied an aggregated technique known as an ensemble using three different fine-tuned CNN models. This study included the ensemble classification method to attain considerably better performance than individual models. We have also studied other architecture of the CNN model, while for the ensemble method, this chapter adopted VGG19, MobileNet, and Xception architecture. The reason for choosing these three architectures is to include different variants of convolution operation techniques applied in CNN models. This variation helps the ensemble method to achieve better performance. Here, the VGG19 uses general convolution operation with no optimization to reduce computational complexity. In contrast, both MobileNet and Xception use depthwise separable convolution, which is much more computationally efficient. However, the technique of depthwise separable convolution is a bit different in MobileNet and Xception.

The newly created dataset of this research contains three different types of insects *Dorsophila melanogaster* (Fruit Fly), *Idiocopus clypealis* (Mango Hopper), and *Sternochetus mangiferae* (Mango Stone Weevil). These species of insects are mostly affecting the mango production in the Asian region [3]. This research collects data from different open sources. Our newly created dataset is one of the main contributions of this research. The dataset contains a different train, test, and validation set with labels.

The contributions of this chapter are as follows:

- Ensemble-based deep learning methods for mango insect classification with improved performances. The ensemble includes fine-tuned VGG19, Xception, and MobileNet.
- The creation of a dataset for mango insect classification consists of the three most dangerous insects- stone weevil, fruit fly, and hopper.
- An intuitive study of deep learning methods for mango insect classification with limited data.

– A study of ensemble methods for performance improvement on insect classification.

The rest of the chapter organized as Sect. 2 presents previous studies related to this research. The ensemble method and pre-trained models fine-tuning are discussed in Sect. 3. Section 4 describes the performance of our proposed method. The last section of this chapter concludes the research and indicates future scopes in mango insect classification.

2 Related Works

Automatic insect classification and detection is an ongoing research topic for the last few years. In order to enhance the classification rate of insects, researchers have studied many techniques using machine learning, pattern recognition, and computer vision. Several machine learning and deep learning techniques have been studied for insect classification and detection. Computer vision and pattern recognition made substantial advances in previous years. Large Scale Visual Recognition Challenge (ILSVRC) based on ImageNet public dataset has been customized as a benchmark for different recognition related problems in computer vision, together with object classification and object identification. Despite the advances in insect detection, there are not many studies on mango insects. This section describes several studies related to dataset collection and insect classification and detection methods in recent years.

Authors of [4] designed fruit fly larvae detection on mango using a hyperspectral imaging technique. They used traditional techniques of image processing to detect and localize fruit fly larvae in mangoes. They claimed 87.7% accuracy of this technique. Xia et al. [5] utilized fine-tuned VGG19 to extract features from insects with a regional proposal network to classify and localize insects. Their proposed method achieved an mAP value of 89.22%. A fusion method of saliency image preprocessing and convolutional neural networks is studied in [6]. They used three different types of preprocessing to extract saliency images and used those images as input for CNN to classify insects with an accuracy of 61.93% in IP102 dataset for insect detection. Researchers of [7] designed a study on insect detection using a saliency map of the image with SIFT-HMAX and LCP feature extraction technique and SVM for classification. The proposed method of [7] can classify within 146.3 s with an accuracy of 85.5%.

Khalifa et al. [8] proposed a transfer learning method for insect classification on IP102 dataset. They used pretrain AlexNet model with an augmentation technique to classify insects into different classes. Their proposed method achieved an accuracy of 89.33% on IP102 test dataset. This method does not provide any comparison with recent advanced deep learning methods. The authors of [9] designed a convolutional network-based feature extraction method to identify the taxonomy of insects. They have used VGG16 network in their study for the feature extraction process. Fuentes et al. [10] proposed a real-time tomato insect detection scheme using deep learning techniques. They combined three different object detection and recognition techniques in their "deep learning meta-architectures." Their work also extended to detect different diseases caused by insects. Besides, the proposed method worked on the localization of disease and insects.

Insect classification using a non-image-based technique is also studied by researchers using machine learning and traditional classification techniques. Phung et al. proposed an insect classification method using the acoustic feature of insects using the bagged tree and kNN classifier [11]. Their proposed classification method on eleven different types of insects achieved an accuracy of 92.3% using a bagged tree while classification accuracy using kNN was 88.5%. Saranwong et al. studied a near-infrared imaging technique for fruit fly larvae [12]. They performed this technique on intact mangoes. The drawback of this work is they worked with one particular type of insects.

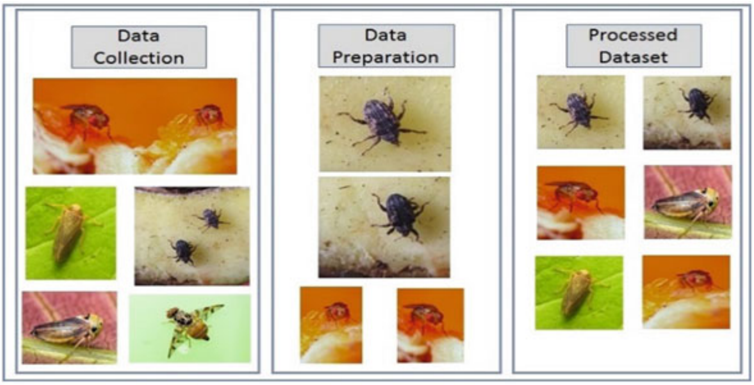
3 Methodology

The methodology section describes the working procedure of our mango insects classification method. This research prepares a specific dataset of mango insects alongside a classification scheme. The first subsection illustrates the overall procedure of dataset preparation. The dataset preparation consists of several steps like data collection, cleaning, labeling, and decomposing, to make the dataset consistent. The next step of the methodology is model preparation. The model preparation step includes data augmentation, model training, and the ensemble of models. Figure 1 presents an overall view of the proposed methodology.

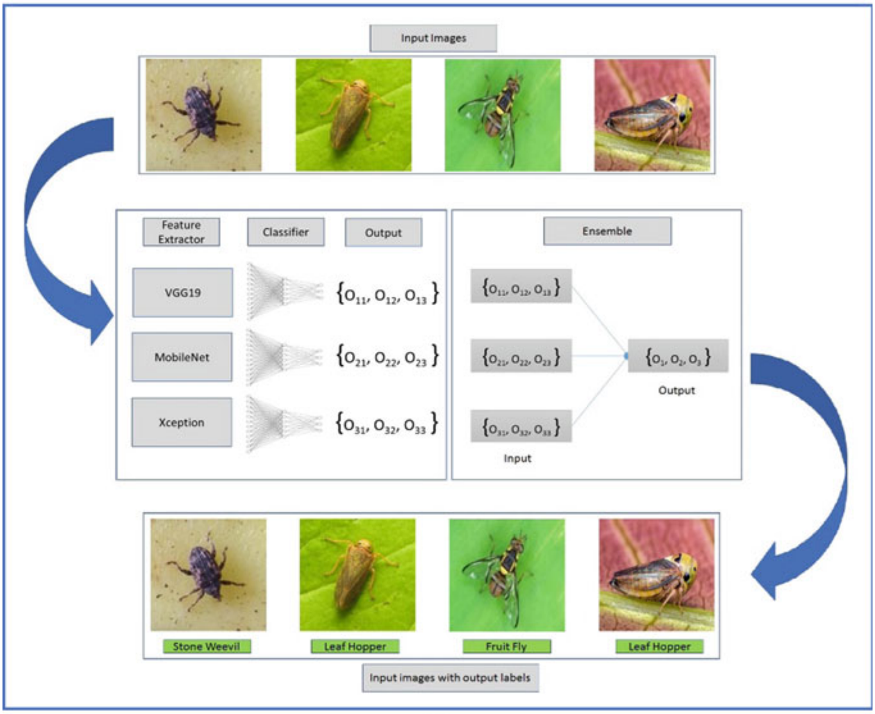
3.1 Dataset Preparation

Dataset preparation is the preliminary task of organizing data for training and evaluating machine learning or deep learning models. Appropriate data preparation aids model to find optimal parameters. Dataset preparation is a complex process to be usable for training deep learning models. The procedure includes defining the problem, and here it is mango insect classification from insect images. The next step towards this procedure is the data collection. The procedure includes sources of data, collection procedure, type of data to collect, and others [13]. This research uses Google image search to collect data from open-source image collections. The unqualified images are removed to maintain consistency in the dataset. The collected data needs to be in a specific format. This research uses a conversion mechanism for formatting images into JPEG format. The images are also resized into a standard dimension of 224×224.

The clean and consistent data are then distributed into train, validation, and test set. The data distribution is a significant part of dataset preparation. The dataset is first



(a)



(b)

Fig. 1 **a** Dataset preparation process. **b** Block diagram of the proposed methodology

Table 1 The distribution of data among three classes (fruit fly, hopper, and stone weevil)

Species	Number of samples
Fruit fly	1215
Hopper	892
Stone weevil	402

divided into two sets- train and test where 85% of the clean data belongs to train set using random selection technique. Then the two datasets are checked for duplicates. The duplicate checking ensures the consistency of data distribution among train and test sets. After that, five percent of the total data is chosen randomly from the train set and make them a validation set. Again, a repetition check is performed on a new train and validation set and taken proper action to remove repetition.

Table 1 shows the distribution of data among different classes. This table presents the overall data distribution. The train set of the whole dataset contains 1215 fruit fly images, 892 hopper images, and 402 stone weevil images. There are 60, 45, and 20 images for fruit fly, hopper, and stone weevil, respectively in the validation set and test set consist of the remaining of the data with 180 images of the fruit fly, 135 hopper images, and 60 images of stone weevil. Table 2 presents the data distribution among train, validation and test set. Figure 2 shows a few instances of different insects belong to this dataset.

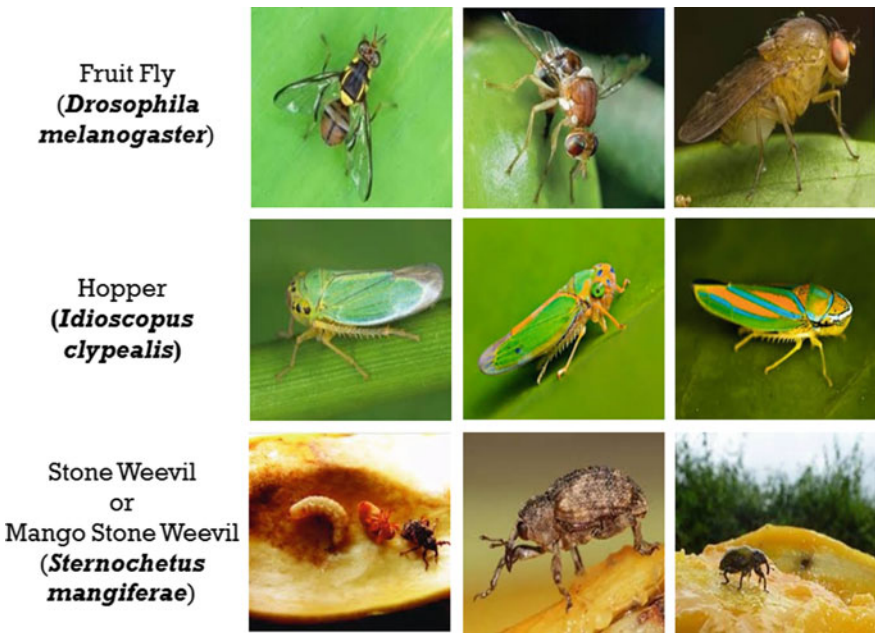


Fig. 2 A few instances of fruit fly, hopper, and mango stone weevil

3.2 Ensemble-Based Classification

Classification is a term popularly used in deep learning, and the machine learning domain refers to the system to categorize data into classes. Herein, we studied mango insect classification using an ensemble technique. The ensemble is a fancy term to represent the aggregation of several methods that work together to find an optimal solution. This study uses an ensemble of three different pre-trained CNN model architecture to classify mango insects. A model that is already trained on a dataset and that weights of that model can be used to solve another problem. This already trained model is known as a pre-trained model. This work uses pre-trained models to get benefits from extensive training of models on large datasets like ImageNet. The three models are Xception, MobileNet, and VGG19. Pre-trained models are not ready to go for every classification; these models require some modification and retraining to be useful for particular purpose. The procedure of retraining a pre-trained model for any particular task is referred to as fine-tuning the model. We have fine-tuned these models as required for insect classification by tuning parameters and hyper-parameters of these models. The fine-tuning of these models are discussed in the next part of this section. In order to compensate with a limited dataset, we have utilized data augmentation techniques to increase the dataset size.

Data Augmentation: Data augmentation is a process to create different variants of the same data to increase diversity in the dataset. Data augmentation is a handy technique to work with limited data. The augmentation of data requires to maintain some constraints to be useful for classification. This work used augmentation techniques to increase dataset size artificially. Most of the mango insects are small in size, thus zooming images to a certain extend is convenient. This study used random zooming of up to 0.2. This random zooming increase diversity while training and reduce model overfitting. Besides, the training procedure also used shearing with randomly chosen value within a range of 0 to 0.1. Flipping is another augmentation technique; we have applied during training.

Moreover, complex image augmentation, like random rotation with zooming, also applied. The pre-trained models require scaled images with pixel values within a range of 0.0 to 1.0. Each image of the dataset is rescaled by dividing the pixel values with the highest intensity value of 255. Fine-tuning of pre-trained models for insect classification is discussed in the next few subsections.

Fine-Tuned VGG19 Model: Training a nineteen weighted layers VGG19 architecture from scratch requires a large amount of data. A trained model with limited

Table 2 Train-validation-test dataset distribution

Dataset	Fruit fly	Hopper	Stone weevil
Train	975	712	322
Validation	60	45	20
Test	180	135	60

data fails to give a good performance. Instead of using a scratch model for limited data, retraining a trained model with vast amounts of data helps to attain considerable performance [14]. This retraining of existing pre-trained is beneficial as these models capable of learning low-level features of any image. Due to a lack of extensive data, we have adopted a pre-trained VGG19 model for classification. The pre-trained VGG19 was previously trained on the ImageNet dataset contains nineteen layers. The pre-trained model can classify images into 1000 classes. However, this research requires images to classify into three categories. To use a pre-trained VGG19 model, we need to fine-tune as our requirement for three-class classification.

The fine-tuning procedure includes several changes in the classifier part of the model. The output layer of the fine-tuned model contains three neurons instead of 1000 neurons of the pre-trained model. This is a three-class (fruit fly, hopper, and stone weevil) classification problem; thus, the fine-tuned model contains three neurons instead of a thousand neurons of the pre-trained model. The model with modified output layer performance is surpassed by a lower number of hidden neurons in hidden layers. After experimenting with several parameters, this research used 512 and 256 neurons on hidden layers with dropout. Besides, dropout facilitates the random presence and absence of each neuron with a certain probability. The model uses the softmax activation function at the output layer and the ReLU activation function at hidden layers. The batch size of the model for training was 64 images of size 224 $\times$ 224. The weight of the convolution part of the pre-trained model is not trained, while the classifier part is randomly initialized and optimized using the mango insect dataset. Figure 3 shows the architecture of the fine-tuned model.

Fine-Tuned MobileNet Model: The MobileNet model uses a different technique for convolution rather than the convolution technique applied in VGG19. The depthwise

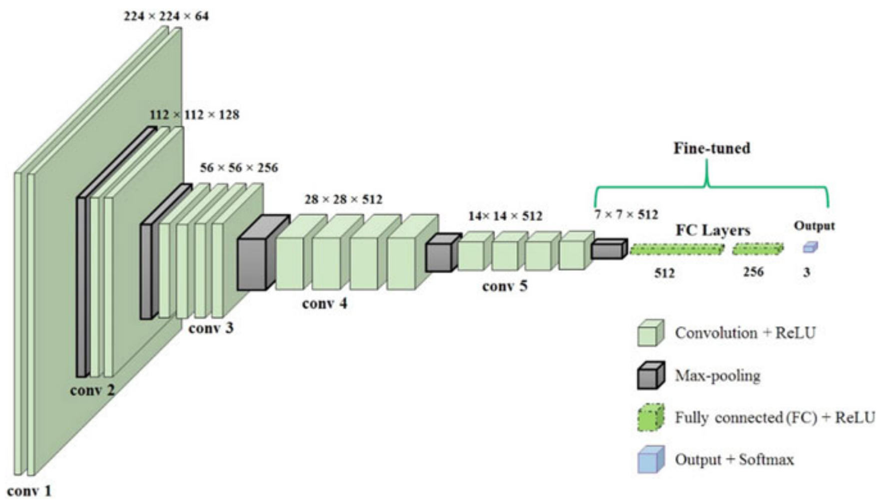


Fig. 3 The architecture of fine-tuned VGG19

Separable Convolution (DSC) technique of the MobileNet model makes it efficient to train with less computing resources without compromising the performance [15]. To utilize the MobileNet pre-trained model for mango insect classification, we need to fine-tune the model.

The fine-tuning process of MobileNet architecture eliminates the classifier head of the architecture and redesigns the classifier part of the model as required to classify mango insects. The experimented dataset contains three different classes thus, the fine-tuned model has three units at the output layer. These three output units or neurons are responsible for classifying an input image to one of the three classes. The average pooling layer of MobileNet architecture is replaced with a Global Average Pooling (GAP) to decrease the fine-tuned model parameters. This layer is followed by two fully-connected layers with 512 neurons on the first hidden layer and 256 neurons on the second. Batch-normalization is applied to speed up the training process. The experiments use dropout to eradicate the effect of over-fitting each hidden layer followed by a dropout layer in this regard. The network uses the ReLU activation function for hidden layers. Figure 4 shows the fine-tuned architecture of the MobileNet model with depthwise convolution visualization.

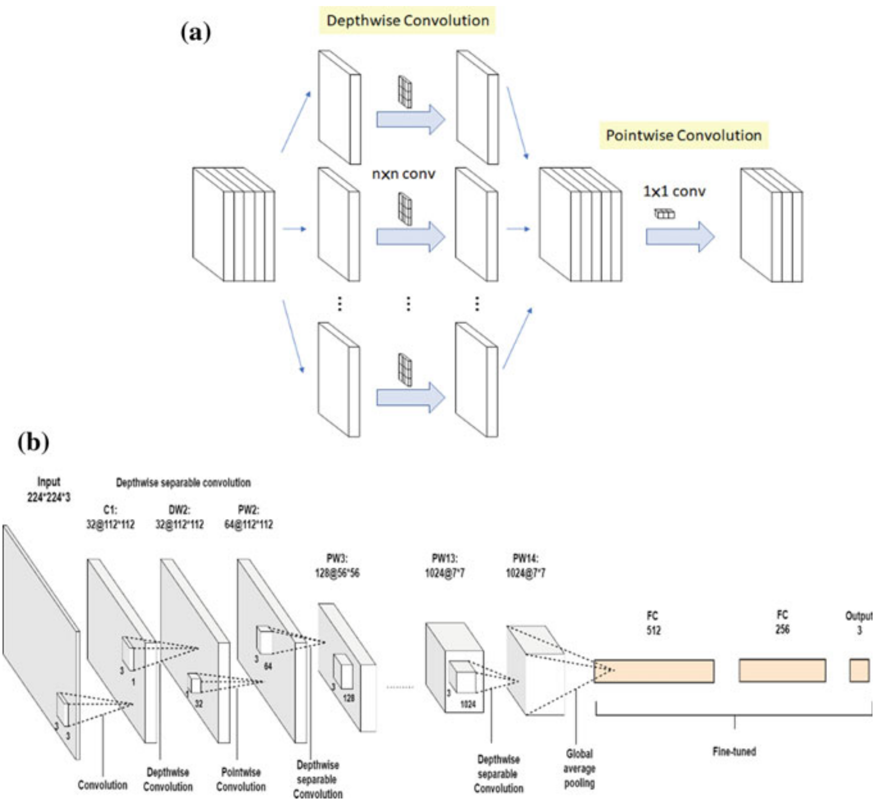


Fig. 4 **a** Depthwise separable convolution of MobileNet and **b** Fine-tuned MobileNet architecture

The output layer uses the softmax activation function. This fine-tuned model uses Adam optimizer to update weights. The early stopping technique is also utilized to stop the training process when validation accuracy remained constant for a few epochs. This technique helps to train the model without overfitting. The learning rate of training is set to 1e-4 at the initial stage of model compilation.

Fine-Tuned Xception Model: The architecture of Xception uses a different form of DSC. The procedure of DSC uses the reverse operation strategy applied in MobileNet DSC. The DSC of Xception performs pointwise convolution, followed by depthwise convolution. This procedure of DSC helps the Xception model to train faster than others with improved performance [16]. Figure 5 illustrates the DSC of Xception with the fine-tuned architecture of Xception.

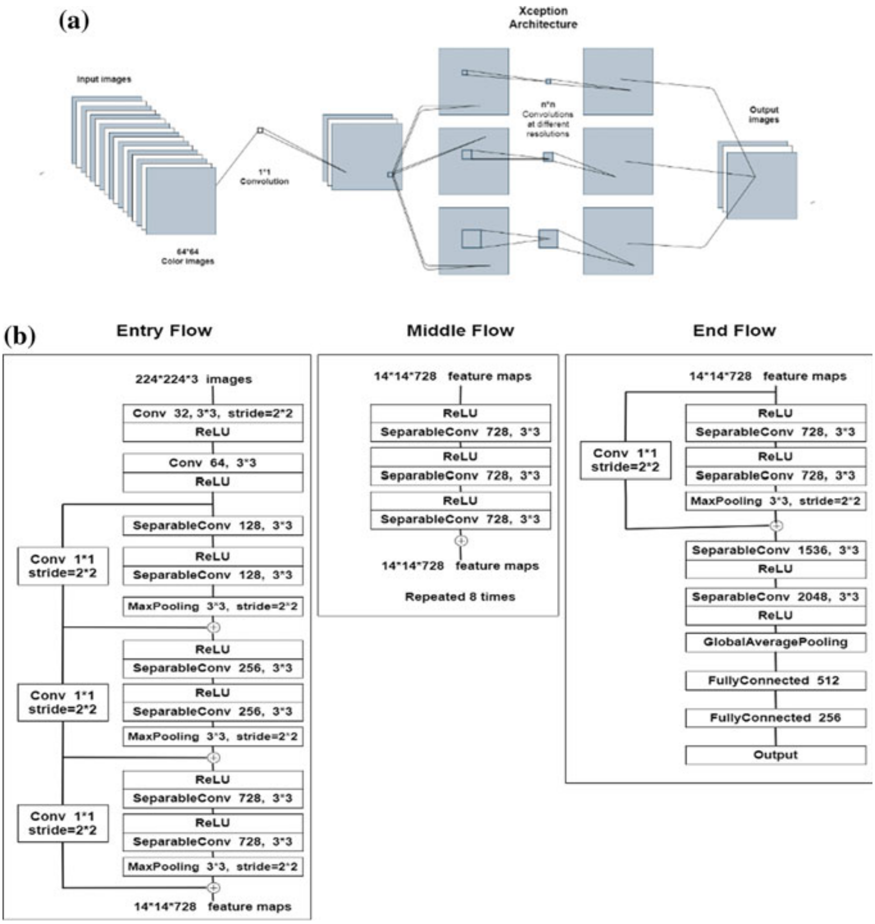


Fig. 5 **a** Depthwise separable convolution of Xception and **b** fine-tuned Xception architecture

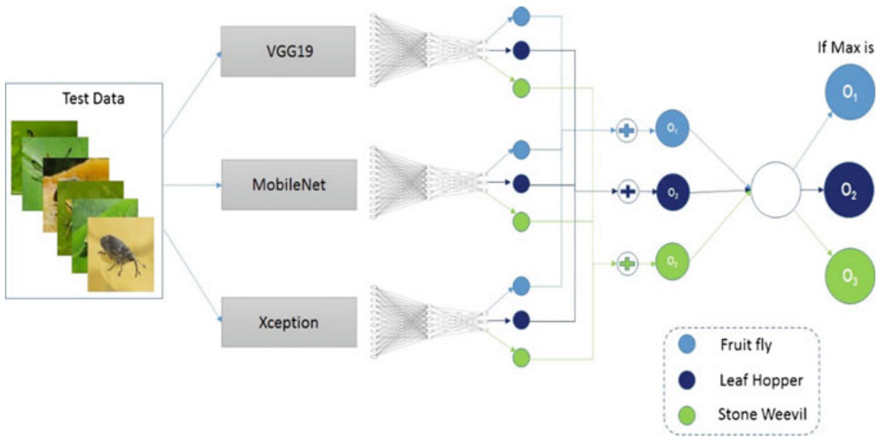


Fig. 6 Ensemble the procedure of the proposed method

The fine-tuning process of the Xception model is similar to two other models with few differences. The output layer of the model has the same number of neurons as the previous two models. The difference between the previous fine-tuning process with this tuned process is that this does not apply GAP as pre-trained already has GAP. We have used two hidden layers with a similar number of neurons as previous models. Batch-normalization is applied after the GAP layer. The hidden layer uses ReLU activation, while the output layer uses softmax activation. Dropout is applied to reduce overfitting. The fine-tuned model also uses Adam optimizer and accuracy as metrics. Early stopping is also utilized as previous models.

The ensemble of Models: The ensemble method is an excellent way of combining results from different methods to achieve improved performance in classification. This proposed method uses the result of the previously mentioned three fine-tuned models. This ensemble method uses the classification output of each model and sums them for each class. The class with the highest sum is the output of the ensemble method. The implemented algorithm uses five digits after the decimal point to minimize tie. Figure 6 shows a graphical illustration of the ensemble method using three fine-tuned model. The ensemble method helps attain better results than a single classifier. This chapter also experimented with other types of ensemble mechanisms such as voting where the final result is the class with the maximum number of votes. However, the summing of probabilities gave a significant performance. Algorithm 1 illustrates the algorithmic classification procedure of mango insects using the ensemble technique.

Algorithm 1 Ensemble technique for classification of mango insects.

Input: Images of mango insects with actual labels as $\{(x_1, y_1), (x_2, y_2), \ldots, (x_m, y_m)\}$, here x_i represents a test image and y_i is the corresponding actual label of the image for $i = 1, 2, 3, \ldots, m$. (m is the total number of test images).

Output: Predicted output (y_pred_i) of images using ensemble classification for $i = 1, 2, 3, \ldots, m$.

begin
Design N models for the intended task $j = 1, 2, 3, \ldots, N$.
for $j = 1$ to N **do**
Train the j^{th} model on the training dataset and find the optimum parameters by utilizing the performance on the validation set.
Store the optimal model as M_j for testing. (M is the collection of N models)

for $i = 1, 2, 3, \ldots, m$ test images **do**
initialization: $max_{prob} = 0$, prob_sum = {0}
for models M: $j = 1$ to N **do**
Give test image x_i as input to the model M_j
for k in 1, 2, …, number of classes (n_c) **do**
$prob\_sum_k$ += value of k^{th} output neuron of M_j
if $prob\_sum_k > max_{prob}$ **then**
$max_{prob} = prob\_sum_k$
$y\_pred_i = k$

Evaluate the performance of ensemble classification using y_i and y_pred_i for i in 1 to m.
end

The algorithm illustrates that the ensemble procedure starts with an image input. Performance evaluation on test images with labels is symbolized as (x_i, y_i) where x_i is the image and y_i is the actual label of the image. Each model outputs the probability value of that image for different classes. The class with the highest probability value is the predicted class of that image. The ensemble technique utilizes the probability value of every class from each model and aggregates the probability of every class in an array. The aggregated probability is stored in *prob_sum* array for the number of total classes (n_c). ith index of *prob_sum* array represents the probability sum of ith class among n_c classes. This aggregation procedure of the ensemble technique overcomes the downfall of using a single model for classification. The maximum value is also calculated while computing the cumulative value of the probability to find the output class.

4 Results and Discussion

The results of mango insect classification are discussed in this section. Besides, this section also presents the result of data collection and preparation of the mango insect dataset. Subsequent subsections present and discuss the results of each fine-tuned model. Section 4.2 demonstrates a detailed performance analysis of the ensemble method.

4.1 Performance of Fine-Tuned Models

VGG19 Performance: Fig. 7 illustrates the performance of the fine-tuned VGG19 model on test data. We can calculate the accuracy of the model from the confusion matrix of the VGG19 model on test data. The accuracy of multiclass classification is calculated using Eq. (1). It is more informative to calculate the precision and recall score of the model on the test dataset. Equations (2) and (3) present the calculating formula for precision and recall. The precision score of the fine-tuned VGG19 model on test data is 89% for fruit fly, 99% for hopper, and 82% stone weevil. The recall score is also known as sensitivity. Recall or sensitivity score of fine-tuned VGG19 is 93%, 92%, and 85% for fruit fly, hopper, and stone weevil accordingly. The overall accuracy of the model on the test dataset is 91%. Table 3 presents the performance metrics of the fine-tuned VGG19 model.

$$\text{Accuracy} = \frac{\text{correctly classified samples } (n_c)}{\text{total number of samples } (n)} \tag{1}$$

$$\text{Precision} = \frac{\text{correctly classified positive samples } \left(n_{tp}\right)}{\text{total number of predicted positive samples } \left(n_{tp+}n_{fp}\right)} \tag{2}$$

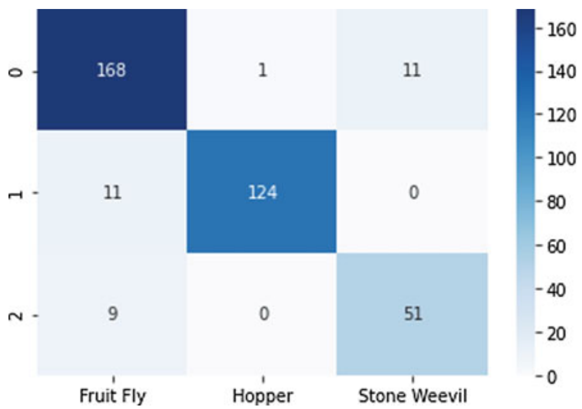


Fig. 7 Confusion matrix of fine-tuned VGG19 model

Table 3 Performance metrics of fine-tuned VGG19 model on mango insect classification test data

Performance metric	Fruit fly	Hopper	Stone weevil	Macro average
Accuracy	0.91			
Precision	0.89	0.99	0.82	0.90
Recall	0.93	0.92	0.85	0.90
F1-score	0.91	0.95	0.84	0.90

$$\text{Recall} = \frac{\text{correctly classified positive samples } (n_{tp})}{\text{total number of actual positive samples } (n_{tp} + n_{fn})} \tag{3}$$

MobileNet Performance: The performance of MobileNet is also calculated on test data. Figure 8 shows the confusion matrix for MobileNet on test data. MobileNet attains 97% precision on fruit fly, 94% on the hopper, and 96% on the stone weevil. The recall score of MobileNet on test data is interesting as it gives 99% on fruit fly and hopper classification. However, MobileNet attains a comparatively low recall score on stone weevil classification with a sensitivity value of 80%. Recall and sensitivity are the same measures with different names. This indicates that the architecture of MobileNet fails to identify stone weevil correctly. Table 4 presents performance metrics.

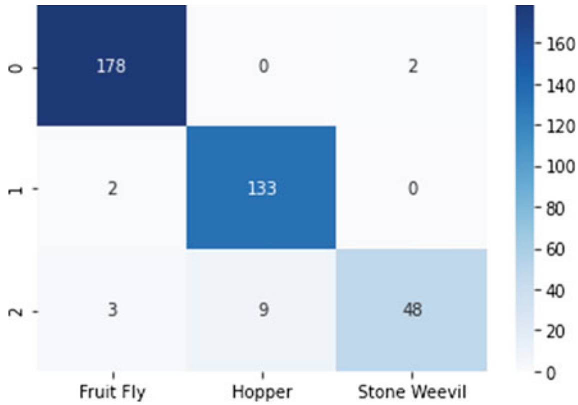


Fig. 8 MobileNet performance on test dataset

Table 4 Performance on test data of fine-tuned MobileNet model

Performance metric	Fruit Fly	Hopper	Stone weevil	Macro average
Accuracy	0.96			
Precision	0.97	0.94	0.96	0.96
Recall	0.99	0.99	0.80	0.92
F1-score	0.98	0.96	0.87	0.94

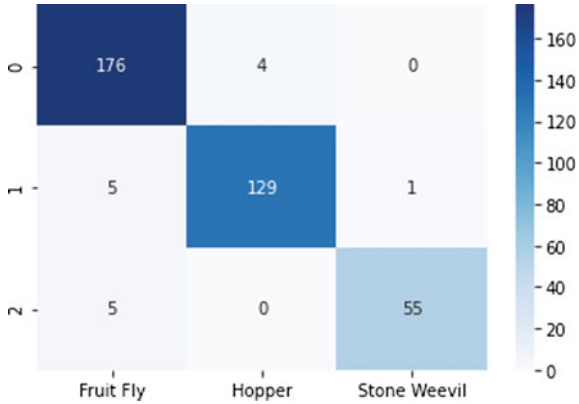


Fig. 9 Confusion matrix of Xception model on test dataset

Table 5 Performance metrics of fine-tuned Xception model on mango insect classification test data

Performance metric	Fruit fly	Hopper	Stone weevil	Macro average
Accuracy	0.96			
Precision	0.95	0.97	0.98	0.97
Recall	0.98	0.96	0.92	0.95
F1-score	0.96	0.96	0.95	0.96

Xception Performance: Fine-tuned Xception gives a better performance than the fine-tuned VGG19 and MobileNet model. The performance accuracy of the Xception model on the test set is 96%. Figure 9 shows the confusion matrix of the Xception model's performance on the test dataset. Table 5 shows the detailed performance metrics for the fine-tuned Xception model. The sensitivity or recall of each class is 98%, 96%, and 92% respectively for fruit fly, hopper, and stone weevil. The precision score of the Xception model on the test dataset is 95% for fruit fly, 98% on stone weevil, and 96% for hopper images.

4.2 *Ensemble Method Performance*

The ensemble method aggregates the results of fine-tuned models and uses the aggregated result for classification. The aggregated results help the ensemble method to boost the performance of classification. This aggregation of results helps to overcome the shortcoming of the individual model. Figure 10 presents the confusion matrix of mango insect classification using an ensemble method. The ensemble method has attained the best precision score and sensitivity score on each model. The precision score of the ensemble method on the test set is as follows 96% for fruit fly, 99% on hopper images, and flawless on stone weevil images. Also, the recall score on the ensemble method is quite good on different images. The recall score of fruit

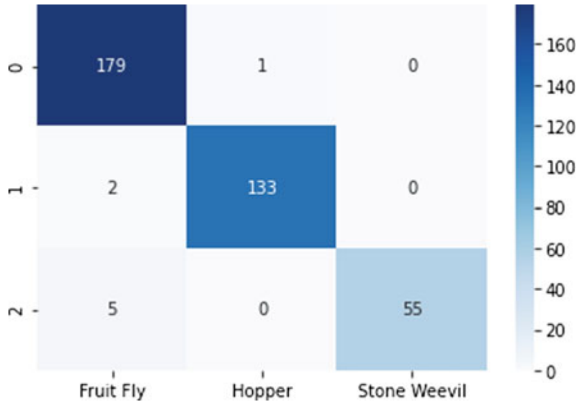


Fig. 10 Performance of ensemble method on test set (Confusion matrix)

Table 6 Performance metrics of ensemble method on test data

Performance metric	Fruit fly	Hopper	Stone weevil	Macro average
Accuracy	0.98			
Precision	0.96	0.99	1.00	0.98
Recall	0.99	0.99	0.92	0.97
F1-score	0.98	0.99	0.96	0.97

fly and hopper images is 99%, while 92% on stone weevil images. The accuracy of the ensemble method on test data is better than each model with an accuracy of 98%. Table 6 shows the performance metrics of ensemble method.

Receiver Operating Characteristic (ROC) curve and Area Under the Curve (AUC): These are two significant metrics for evaluating classification performance [17]. Figure 11 represents the ROC curve with the AUC value of different classes for ensemble classification. The figure also represents the micro and macro average of the ROC curve with the AUC score. Hopper images achieved the best AUC score of 0.99, and stone weevil has the least AUC score. Both micro and macro average [18] AUC score of the ensemble method is 0.98. The interpretation of the AUC score indicates that the higher the classification model's value, the better the model is for classification. From this interpretation, it is evident that this proposed method performs well on mango insect classification.

Precision-Recall (PR) curve: The PR curve presents the precision and recall value of the classification model at different thresholds. The PR curve is more important for methods that work on an imbalanced dataset. The imbalance between the number of images in the largest class and smallest class in our dataset is not significantly high. PR curve gives significantly better intuition on the moderate class imbalance. Figure 12 shows the PR curve of our proposed method.

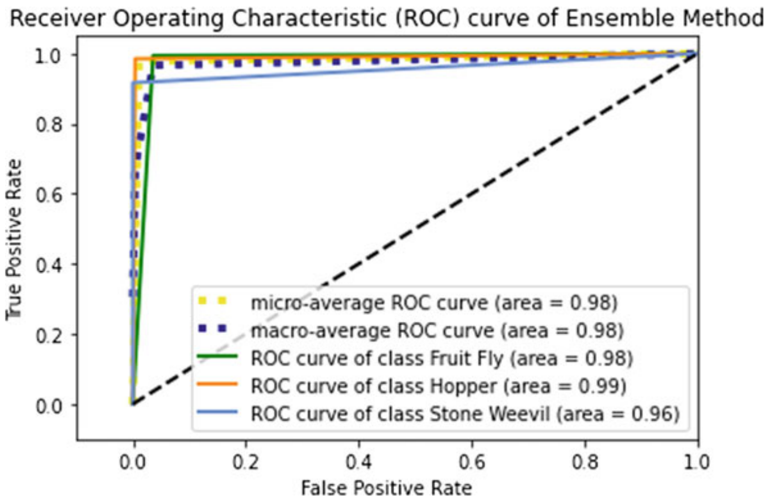


Fig. 11 ROC curve with AUC score of ensemble method

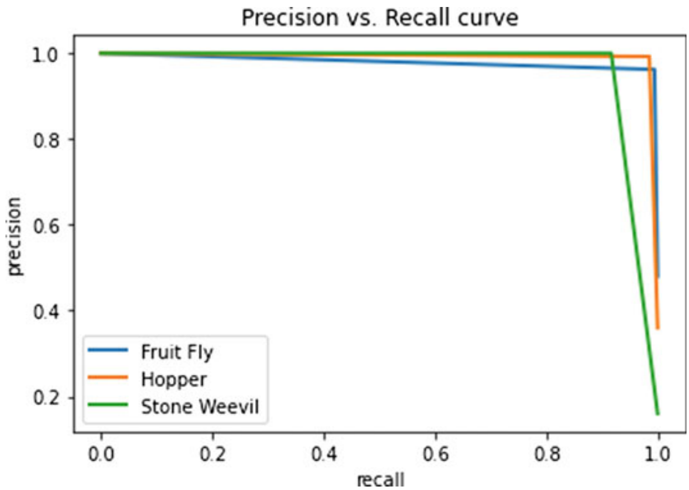


Fig. 12 Precision versus Recall curve of ensemble method

5 Conclusion

The adverse of insects in mango production cause substantial economic loss. The use of pesticides without knowing the category of pests can cause severe damage to the growth of mango. This chapter proposed a mango insect classification scheme using an ensemble of CNN models. Besides, this study developed a dataset of mango insects to aid automated mango insect classification. The dataset contains three thousand clean images with labels distributed into train, validation, and a test set of three different kinds of mango insects named mango stone weevil, mango hopper, and fruit fly. This research also presented a study of ensemble techniques to classify mango insects. The ensemble technique used three well-performed fine-tuned CNN models to boost the performance of the ensemble technique with an accuracy of 98%. This work will be extended to increase the dataset size with several types of mango insects in the future. It will be interesting to use other techniques to detect insects in real-time.

References

1. Mugure, M.C.: Economic assessment of losses due to fruit fly infestation in mango and the willingness to pay for an integrated pest management package in embu district, Kenya. J. Chem. Inf. Model. (2013)
2. Rami Reddy, P.V., Gundappa, B., Chakravarthy, A.K.: Pests of Mango. In: Pests and Their Management (2018)
3. Pablo, F.B., Simpao, J.T., Arcelo, M.M.: Survey of major insect pests of mango and their natural enemies in Region 11 South Eastern Mindanao [Philippines]. In: Proceedings of the thirty third Anniversary & Annual Scientific Meeting of the Pest Management Council of the Philippines, Inc. College, Laguna (Philippines) (2002)
4. Haff, R.P., Saranwong, S., Thanapase, W., Janhiran, A., Kasemsumran, S., Kawano, S.: Automatic image analysis and spot classification for detection of fruit fly infestation in hyperspectral images of mangoes. Postharvest Biol. Technol. (2013)
5. Xia, D., Chen, P., Wang, B., Zhang, J., Xie, C.: Insect detection and classification based on an improved convolutional neural network. Sensors (Switzerland) (2018)
6. Nanni, L., Maguolo, G., Pancino, F.: Insect pest image detection and recognition based on bio-inspired methods. Ecol. Inform. (2020)
7. Deng, L., Wang, Y., Han, Z., Yu, R.: Research on insect pest image detection and recognition based on bio-inspired methods. Biosyst. Eng. (2018)
8. Khalifa, N.E.M., Loey, M., Taha, M.H.N.: Insect pests recognition based on deep transfer learning models. J. Theor. Appl. Inf. Technol. (2020)
9. Valan, M., Makonyi, K., Maki, A., Vondráček, D., Ronquist, F.: Automated taxonomic identification of insects with expert-level accuracy using effective feature transfer from convolutional networks. Syst. Biol. (2019)
10. Fuentes, A., Yoon, S., Kim, S.C., Park, D.S.: A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition. Sensors (Switzerland) (2017)
11. Phung, Q.V., Ahmad, I., Habibi, D., Hinckley, S.: Automated insect detection using acoustic features based on sound generated from insect activities. Acoust. Aust. (2017)
12. Saranwong, S., Haff, R.P., Thanapase, W., Janhiran, A., Kasemsumran, S., Kawano, S.: Short communication a feasibility study using simplified near infrared imaging to detect fruit fly larvae in intact fruit. J. Near Infrared Spectrosc. (2011)
13. Roh, Y., Heo, G., Whang, S.E.: A survey on data collection for machine learning: a big data—ai integration perspective. IEEE Trans. Knowl. Data Eng. (2019)
14. Hasan, M.M., Islam, N., Rahman, M.M.: Gastrointestinal polyp detection through a fusion of contourlet transform and Neural features. J. King Saud Univ. Comput. Inf. Sci. (2020)
15. Howard, H.A.A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M.: Mobilenets: efficient convolutional neural networks for mobile vision applications. In: Computer Vision and Pattern Recognition (2009)
16. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In: Proceedings—30th IEEE conference on computer vision and pattern recognition, CVPR 2017 (2017)

17. Hand, D.J., Till, R.J.: A Simple generalisation of the area under the ROC curve for multiple class classification problems. Mach. Learn. (2001)
18. Asch, V.V.: Macro-and micro-averaged evaluation measures. Belgium: clips (2013)

Implementation of a Convolutional Neural Network for the Detection of Tomato-Leaf Diseases

Md. Ismail Jabiullah, Shovon Ahammed, Mostafizur Rahman, Mahedi Hasan, and Kawser Ahmed

1 Introduction

Bangladesh is an agricultural country and here, agriculture is the backbone of its economy. It produces a tomato every year. Most of the people in Bangladesh live in the village. And most of the village people are farmers. Their life depends on agriculture. They work from dawn to dusk to produce the crop. Tomato is one of them. It takes almost 3 months to harvest. So, it is a very long time. The tomato is full of lutein and lycopene. It has got many health benefits. In these 3 months, the farmers left no stone unturned to produce a good crop. But due to disease, they do not get the expected production. Not only the farmer suffers but also the whole economy of our country suffers due to a loss of production. Again, Bangladesh is a densely populated country that is why these losses are irrecoverable. In this work, nine tomato-leaf diseases are taken into consideration. If these can be determined earlier then proper measures can be taken to protect the production.

But it is hard to detect them at an earlier stage with our eyes. Because there are huge similarities between most of the diseases (see Fig. 1). But there are significant

Md. Ismail Jabiullah (✉) · S. Ahammed · M. Rahman · M. Hasan · K. Ahmed
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh
e-mail: drismail.cse@diu.edu.bd

S. Ahammed
e-mail: shovon15-7671@diu.edu.bd

M. Rahman
e-mail: mostafizur15-7764@diu.edu.bd

M. Hasan
e-mail: mahedi15-7763@diu.edu.bd

K. Ahmed
e-mail: kawser15-7973@diu.edu.bd

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_6

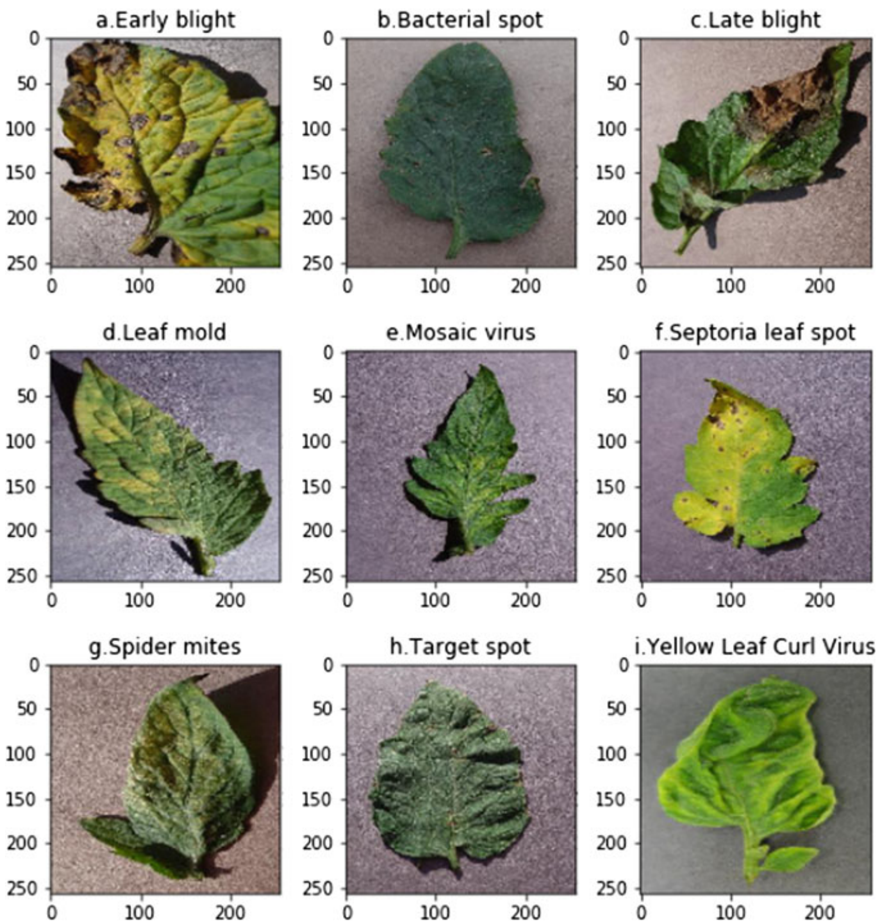


Fig. 1 Images of tomato-leaf diseases

individual features which help to detect the actual disease. Many works have been done on tomato leaf disease detection. But most of them have worked with only a few numbers of diseases.

Now-a-day's technology is taking big steps to solve human problems. Deep learning is an efficient technology that is widely used in many such applications. That is why deep learning algorithms have been used in this approach. The convolutional neural network (CNN) is a deep learning strategy that is used to address in this work to solve the tomato-leaf disease recognition problem.

The remaining parts of the chapter are organized as follows.

Section 2 describes the literature review. Section 3 presents research methodology including the implementation of CNN. Section 4 illustrates the research results and finally, Sect. 5 concludes the chapter.

2 Related Works

There are many techniques available to detect and classify agricultural diseases. Most of them are based on image processing techniques. The key step is feature extraction from the image.

Sharath et al. [1] have detected bacterial blight in pomegranate. They have used images to perform the task. For segmentation, they have used the GrabCut segmentation technique. After segmentation, they have used the Gaussian mixture model (GMM) on the data. On the feature extraction part, they have one sub-section called edge detection. Canny edge detection has been used to detect the affected edges of pomegranate. Khirade and Patil [2] have analyzed plant disease using an image processing technique. They have discussed four steps. In the image segmentation part, they have discussed three methods. First, they have discussed image segmentation using boundary and spot detection algorithm. Then they have discussed K-means clustering and the Otsu threshold algorithm. Devraj et al. [3] have recognized plant diseases using an image processing technique. In the image preprocessing section, they have removed the noise, which results in the enhancement of the image information. They have used K-means clustering in image segmentation. At last, a random forest classifier has been used for classification. Shaikh and Dhole [4] have detected an unhealthy region of a citrus leaf using an image processing technique. A gray-level co-occurrence matrix (GLCM) has been used to extract features. Finally, they have used Hidden Markov Model for classification. They have classified anthracnose with an accuracy of 84.21%, canker with an accuracy of 85.71%, citrus greening with an accuracy of 78%, overwatering with an accuracy of 82.5%. Zhang et al. [5] made their approach with an active contour model to segment cotton leaf disease. They calibrate the deviation with the help of the penalty function. The contour model got smooth closed contour curves of the diseased leaf. Yuan et al. [6] detected anthracnose disease of tea plants with the help of hyperspectral imaging. The diseased portion of the leaf reflects a spectrum of different levels. Their proposed HIS method accurately detected anthracnose. Tian et al. [7] segmented the tomato-leaf image with the clustering number of the K-means algorithm. They got an average F_1-score of 0.765 and entropy of 0.643. Sharif et al. [8] have used weighted segmentation and feature selection to detect and classify citrus diseases. They have worked with six types of citrus diseases. They have classified the diseases with support vector machines based on selected features. They got an average classification accuracy of 92.45%. Pantazi et al. [9] have made a one-class classifier to detect leaf disease by image processing. Their model has classified diseased leaves with an accuracy of 95%. Ma et al. [10] proposed a deep convolutional neural network (DCNN) model for cucumber disease recognition. It has been found that AlexNet performed better than DCNN in their approach.

Dhingra et al. [11] have detected and classified leaf disease by neutrosophic computer vision approach. They have pre-processed the data using contrast limited adaptive histogram equalization (CLAHE) algorithm. They selected a region of interest (ROI) using the neutrosophic technique. They have used nine classifiers.

The best accuracy of those classifiers has been achieved up to 98.4%. Gensheng et al. [12] have classified tea leaf diseases by low shot learning. They have extracted tea leaf features from color and texture and segmented by SVM. The data have been augmented using improved conditional deep convolutional generative adversarial networks (C-DCGAN) and used to train VGG16, which has exhibited a maximum accuracy of 90%. Ozguven et al. [13] have shown a faster approach with R-CNN to detect and classify leaf spot disease in sugar beet, which has resulted in an accuracy of just below 95.5%. Adeel et al. [14] have detected grape leaf diseases by multiple features fusion. They removed the noise from the fusion using Neighborhood Component Analysis (NCA). They got segmentation accuracy of 90% and classification of the accuracy of 92%. Coulibaly et al. [15] have proposed an approach for recognition of mildew disease in pearl millet using transfer learning, whose accuracy has been 95%.

3 Methodology

The approach is to detect nine types of tomato disease efficiently. The convolutional neural network (CNN) has been used to perform the task. CNN performs the classification of an image with the help of the convolution, pooling, and fully connected layer. The proposed approach is with 9 types of diseases. A typical implementation of CNN is not efficient enough. Keras has been used in this approach. Keras is a neural network library. It is open-source and written in Python programming language. The dataset is collected from Kaggle. A sequential model has been used in this approach. For convolution one layer of output works as an input layer for the next layer. So, that is why the sequential model has been used in this approach. The activation function is a very important factor for this work. Without the activation function, the model would produce a linear function which nothing but a one-degree polynomial. The model needs to learn features from the image. Though a linear function is easy, it is not useful for complex operations. But non-linear function can perform the task very easily.

There are four convolutions in this approach. Every convolution has used max-pooling. The proposed methodology is divided into three parts. At first, the input of the image. Then feature learning with the help of different layers. Finally, classification with the help of flattening and fully connected layer. As the approach is for multiclass classification so, soft-max classifies the image into a different class according to feature (see Fig. 2).

The performance of the model largely depends on the activation function. In this approach, three activation functions have been considered. All of them have some benefits and issues. They are sigmoid, tanh, and ReLu.

Sigmoid activation function got a range between 0 and 1. It has got a gradient vanishing problem. Since the output of the sigmoid function is not zero centered, it makes the optimization harder. Again, sigmoid got slow convergence. It has got a saturation issue. For this reason, it destroys the gradients.

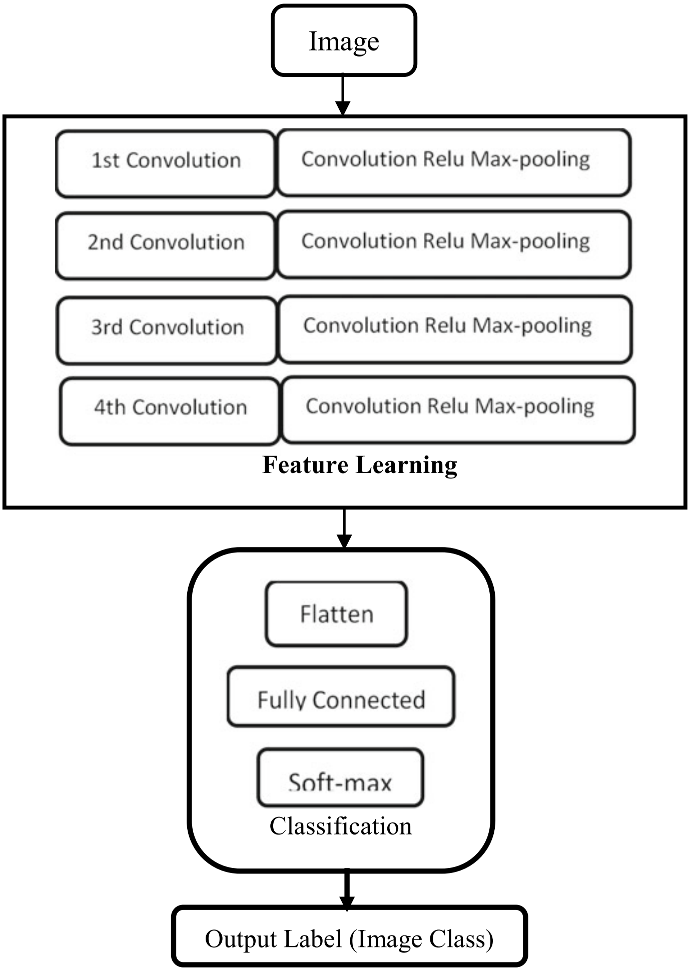


Fig. 2 Flow diagram of the approach

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{1}$$

Since tanh function has got an output range between −1 to +1. So, its output is zero centered. Compared to sigmoid optimization is easier for tanh. But the main problem is it has a gradient vanishing problem.

$$\tan h(x) = \frac{2}{1 + e^{-2x}} - 1 \tag{2}$$

For activation in each hidden layer, we have used Relu (Rectified Linear Unit). It is very simple and efficient. It has got much better convergence compared to tanh. For Relu

$$R(x) = \max(0, x)$$
$$\text{if } x < 0, \tag{3}$$

$$R(x) = 0$$
$$\text{if } x > 0, \tag{4}$$

$$R(x) = x \tag{5}$$

Figure 3 represents the graph of the activation function. As the sigmoid range between zero to one. So, the graph shows that the curve is starting from zero and going up to one. The graph is "S" shaped. The horizontal points for hyperbolic graph are −5.0, −2.5, 0.0 and 5.0. And the vertical points for the hyperbolic graph are −1.0, −0.75, −0.50, −0.25, 0.00, 0.25, 0.50, 0.75 and 1.00. Relu is the one that has been used for the model is looking linear for positive values and if there is a negative value it transforms into zero.

Pooling is another important aspect of this approach. There are two most useful pooling techniques available. They are max-pooling and average pooling. Average pooling takes the average of shape.

The model contains max-pooling with the shape of (2, 2). Max-pooling reduces the dimension and helps to learn the feature. The input contains lots of features which are not necessary for the disease classification task. The max-pooling has been used in all the convolution to reduce dimension. This pooling takes the maximum value in the matrix. This technique uses another parameter called stride. Stride is the number of pixels that move at each iteration.

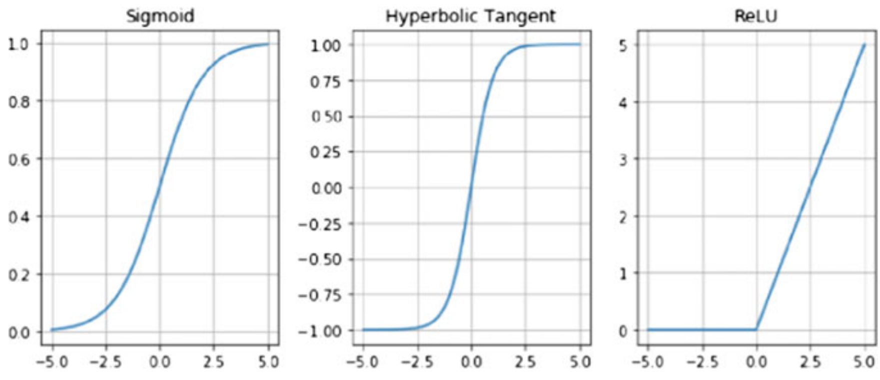


Fig. 3 Graph of activation functions

Figure 4 represents the max-pooling of (2 × 2). According to max-pooling, the array of shape (4 × 4) can give 4 maximum number of values. So, the array of (4 × 4) has been converted to an array of (2 × 2) with the feature (Table 1).

The model has a total of 3,477,066 parameters and all of them are trainable. Each layer has a max-pooling shape of (2, 2). Max-pooling reduces the dimension and helps to learn the feature. It takes the maximum value from a particular section of the pixel. The sigmoid optimizer has been used with a clip value of 0.5. Image augmentations have been used to make the best use of the training dataset.

Table 2 represents all the parameters along the values used in this approach. The dataset is divided into two categories: a training set and a testing set. In the training set, images have been used to train the model. Inside the training set, there are ten classes; nine of them are diseases but one of them is a healthy leaf.

There are a total of 11,803 images for training the model. Among them, the Bacterial spot has 1700, Early blight has 798, Late blight has 1526, Leaf mold has 760, Septoria leaf spot has 1415, Spider mites have 1339, images for training. The target spot has 1124 images, the mosaic virus has 297 images for training. The yellow leaf curl virus has 1573 images for training. The rest number of images are for a healthy leaf. So, we have used almost twelve thousand images for training our model and all of them are RGB images. Figure 5 represents the training data set.

To accomplish the approach validation part is also important. To validate the model, 3208 images have been used. The Fig. 6 represents the visualization of the validation data set. We have used a total of 3208 images to test our model for 9

12	20	30	0
8	12	2	0
34	70	37	4
112	100	25	12

$\xrightarrow{2\times 2 \text{ Max-pool}}$

20	30
112	37

Fig. 4 Max-pooling of 2 × 2

Table 1 Tomato disease classifier

Layer type	Output shape	Parameter
Conv2D	(0, 148, 148, 64)	1792
Maxpooling2D	(0, 74, 74, 64)	0
Conv2D	(0, 72, 72, 64)	36,928
Maxpooling2D	(0, 36, 36, 64)	0
Conv2D	(0, 34, 34, 128)	73,856
Maxpooling2D	(0, 17, 17, 128)	0
Conv2D	(0, 15, 15, 128)	147,584
Maxpooling2D	(0, 7, 7, 128)	0
Flatten	(0, 6272)	0
Dropout	(0, 6272)	0
Dense	(0, 512)	3,211,776
Dense	(0, 10)	5130

Table 2 Image augmentation parameters

Parameter name	Value
Rotation range	40
Width shift range	0.2
Height shift range	0.2
Rescale	1.0/255.0
Shear range	0.2
Zoom range	0.2
Horizontal flip	True
Fill mode	Nearest

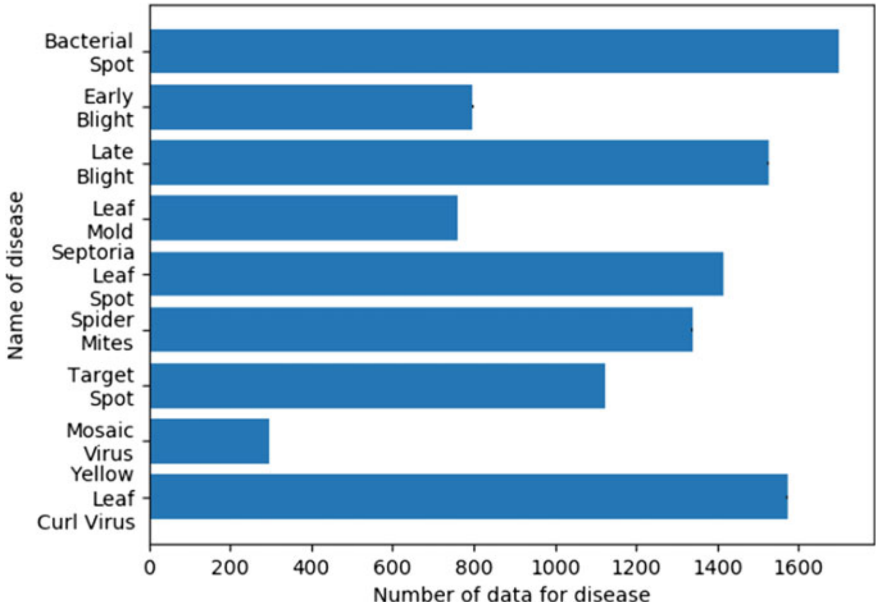


Fig. 5 Horizontal bar chart of training data

diseases. Among them, the Bacterial spot has 423 images for testing. Early blight has 198 images for testing. Late blight has 379 images for testing. Leaf mold has 188 images for testing. Septoria leaf spot has 352 images for testing. Spider mites have 333 images for testing. The target spot has 278 images for training. The mosaic virus has 72 images for testing. The yellow leaf curl virus has 669 images for testing. The rest number of images is for a healthy leaf. So, we have used more than three thousand images for testing our model and all of them are RGB images.

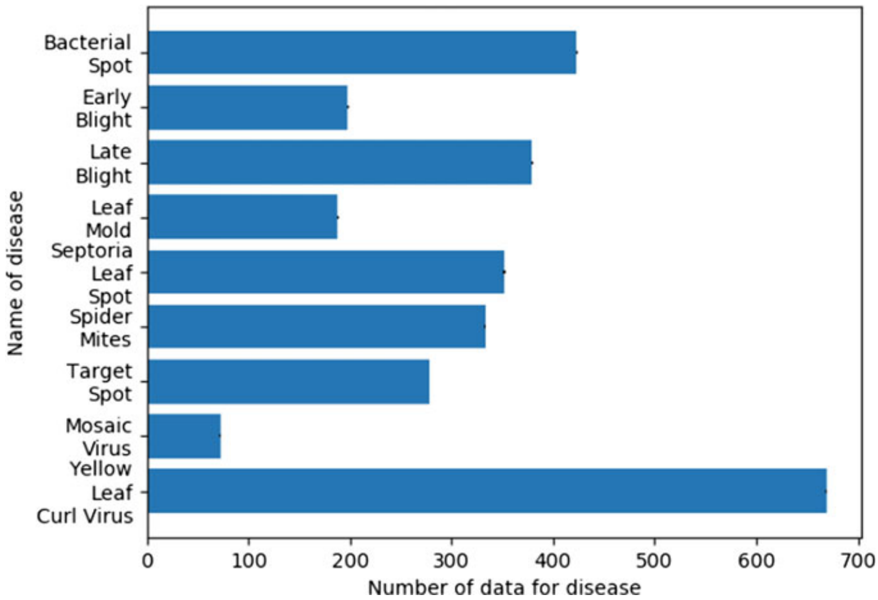


Fig. 6 Horizontal bar chart of validation data

4 Results Analysis

After completing 50 epochs, the model got a maximum accuracy of 90% on the training set and an accuracy of 93% on the validation set. This high accuracy means the model is very well trained, and it is capable enough to detect those diseases accurately. The accuracy of training and validation was increasing linearly though there is some decrease as well for some points. But the accuracy has increased constantly. Training accuracy is more linear compared to validation accuracy. After the first few epochs, the validation accuracy went down drastically but after that, it kept increasing (Fig. 7).

With each epoch, the loss for training and validation kept on decreasing. There is some increment of loss at some points. But for the maximum number of epochs, the loss kept on decreasing. Minimal loss is an indication of a good model. The trained model has been tested with some data which is unknown for the model. There are a total of ten classes in the model. So the model predicts those classes.

Then the model has been used to predict new data. For the new dataset, the Bacterial spot has got a precision of 0.95, recall of 0.88, F_1-score of 0.91 and support of 40. Early blight has got a precision of 0.89, recall of 0.91, F_1-score of 0.9 and support of 45. Late blight acquired precision of 0.89, recall of 0.89, F_1-score of 0.89 and support of 35. Leaf curl scored precision of 0.86, recall of 0.90, F_1-score of 0.88 and support of 40. Leaf mold has got a precision of 0.88, recall of 0.88, F_1-score of 0.88 and support of 50. Mosaic acquired precision of 0.91, recall of 0.88, F_1-score

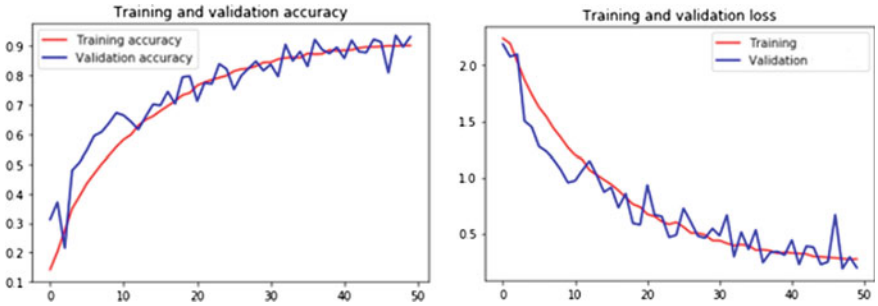


Fig. 7 Training, validation accuracy and training, validation loss

of 0.90 and support of 60. Septoria spot scored precision of 0.93, recall of 0.87, F_1-score of 0.90 and support of 30. Spider mites have got a precision of 0.90, recall of 0.88, F_1-score of 0.89 and support of 40. And Target spot acquired precision of 0.87, recall of 0.87, F_1-score of 0.87 and support of 45.

Among the diseased, the model has got the highest precision of 95% for bacterial blight and the lowest precision of 86% for leaf curl. Early blight has got the highest recall of 91% and lowest of 87% for both septoria and target spot. The highest F_1-score of 91% scored by bacterial blight and the lowest F_1-score of 87% is scored by target spot. Mosaic has got maximum support of 60 and late blight have got the lowest support of 35.

Precision is the value of true positive divided by the summation of the true positive and false positive. The **recall** is true positive divided by the summation of the true positive and false positive. F_1-score is the measure of prediction accuracy (Fig. 8; Table 3).

The confusion matrix visualizes the prediction with the graph. The horizontal axis shows predictions and the vertical axis shows the actual result. The diagonal portion of the confusion matrix is True Positive for the classes.

5 Conclusion

The ultimate aspect of this study is to detect 9 kinds of tomato-leaf diseases using a convolutional neural network. In this chapter, different methods of leaf disease detection have been analyzed. The investigated CNN model consists of 4 convolutional layers. After analyzing different activation functions ReLu has been used for activation of the hidden layer. As the work is about multi-class classification, so we used a soft-max function. A total of 11,803 images has been used to train the model. Image augmentation has been used to make the best use of the dataset. The model has a maximum accuracy of 93%. For the elaborate study, we would like to use a database containing more data and to compare our results with the results of the state-of-the-art DNN models. As this model gives high performance, we expect

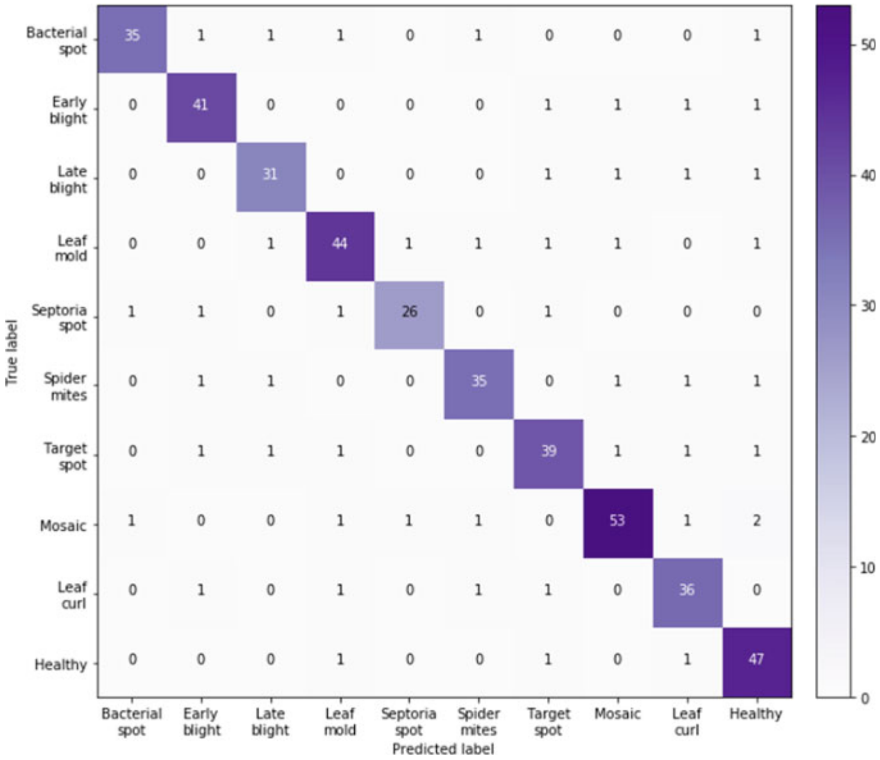


Fig. 8 confusion matrix of prediction

Table 3 Clasification report

Class	Precision	Recall	F_1-score	Support
Bacterial spot	0.95	0.88	0.91	40
Early blight	0.89	0.91	0.90	45
Late blight	0.89	0.89	0.89	35
Leaf curl	0.86	0.90	0.88	40
Leaf mold	0.88	0.88	0.88	50
Mosaic	0.91	0.88	0.90	60
Septoria spot	0.93	0.87	0.90	30
Spider mites	0.90	0.88	0.89	40
Target spot	0.87	0.87	0.87	45

that these results will help for further research. Moreover, it will help to develop an image application through the application, general people can classify it.

References

1. Sharath, D.M., Kumar, S.A., Rohan, M.G., Prathap, C.: April. Image based plant disease detection in pomegranate plant for bacterial blight. In: 2019 International Conference on Communication and Signal Processing (ICCSP), pp. 0645–0649. IEEE (2019)
2. Khirade, S.D., Patil, A.B.: February. Plant disease detection using image processing. In: 2015 International Conference on Computing Communication Control and Automation, pp. 768–771. IEEE (2015)
3. Devaraj, A., Rathan, K., Jaahnavi, S., Indira, K.: Identification of plant disease using image processing technique. In: 2019 International Conference on Communication and Signal Processing (ICCSP), pp. 0749–0753. IEEE (2019)
4. Shaikh, R.P., Dhole, S.A.: Citrus leaf unhealthy region detection by using image processing technique. In: 2017 International conference of Electronics, Communication and Aerospace Technology (ICECA), Vol. 1, pp. 420–423. IEEE (2017)
5. Zhang, J., Kong, F., Wu, J., Han, S., Zhai, Z.: Automatic image segmentation method for cotton leaves with disease under natural environment. J. Integr. Agric. **17**(8), 1800–1814 (2018)
6. Yuan, L., Yan, P., Han, W., Huang, Y., Wang, B., Zhang, J., Zhang, H., Bao, Z.: Detection of anthracnose in tea plants based on hyperspectral imaging. Comput. Electron. Agric. **167**, 105039 (2019)
7. Tian, K., Li, J., Zeng, J., Evans, A., Zhang, L.: Segmentation of tomato leaf images based on adaptive clustering number of K-means algorithm. Comput. Electron. Agric. **165**, 104962 (2019)
8. Sharif, M., Khan, M.A., Iqbal, Z., Azam, M.F., Lali, M.I.U., Javed, M.Y.: Detection and classification of citrus diseases in agriculture based on optimized weighted segmentation and feature selection. Comput. Electron. Agric. **150**, 220–234 (2018)
9. Pantazi, X.E., Moshou, D., Tamouridou, A.A.: Automated leaf disease detection in different crop species through image features analysis and one class classifiers. Comput. Electron. Agric. **156**, 96–104 (2019)
10. Ma, J., Du, K., Zheng, F., Zhang, L., Gong, Z., Sun, Z.: A recognition method for cucumber diseases using leaf symptom images based on deep convolutional neural network. Comput. Electron. Agric. **154**, 18–24 (2018)
11. Dhingra, G., Kumar, V., Joshi, H.D.: A novel computer vision based neutrosophic approach for leaf disease identification and classification. Measurement **135**, 782–794 (2019)
12. Hu, G., Wu, H., Zhang, Y., Wan, M.: A low shot learning method for tea leaf's disease identification. Comput. Electron. Agric. **163**, 104852 (2019)
13. Ozguven, M.M., Adem, K.: Automatic detection and classification of leaf spot disease in sugar beet using deep learning algorithms. Physica A **535**, 122537 (2019)
14. Adeel, A., Khan, M., Sharif, M., Azam, F., Shah, J., Umer, T., Wan, S.: Diagnosis and recognition of grape leaf diseases: An automated system based on a novel saliency approach and canonical correlation analysis based multiple features fusion. Sustain. Comput. Inform. Syst. **24**, 100349 (2019)
15. Sun, S., Li, C., Paterson, A., Chee, P., Robertson, J.: Image processing algorithms for infield single cotton boll counting and yield prediction. Comput. Electron. Agric. **166**, 104976 (2019)

A Multi-Plant Disease Diagnosis Method Using Convolutional Neural Network

Muhammad Mohsin Kabir, Abu Quwsar Ohi, and M. F. Mridha

1 Introduction

Plant disease is a common threat to the quality and quantity of global agricultural production. Disastrous plant disease enhances the current shortage of the food supply in which at least 0.8 billion people are inadequately fed [1]. Additionally, it is a significant threat to food security, where the number of consumers is increasing daily. To reduce harm, we must identify the disorder immediately. In particular, viral plant disorders have no solutions, and they spread rapidly. Thus, transited plants must be dispelled instantly to abstain from secondary infections. To remove infected plants immediately, the diagnosis of the infected plant is the most significant task.

The diagnosis and recognition of plant diseases play a vital role in ensuring the high quality and quantity of food production. Automated plant disease diagnosis is an inevitable research topic, and it is being studied with various researchers' approaches. The leaves of plants are the common element of plant disease detection, and the symptoms of most diseases start to appear on the leaves [2]. Therefore, identifying disease using leaf images is the most general method for researchers.

Farmers often use the internet or experienced gardeners' opinions to diagnose plant diseases. Sometimes, farmers take a small portion of an infected plant or picture it to the local agricultural health center. The diagnosis of plant disorders by manual optical observation of plant leaves is often time-consuming and laborious. Furthermore, inaccurate diagnosis is prevalent. In addition, experienced agriculturists and

M. M. Kabir · A. Q. Ohi · M. F. Mridha (✉)
Bangladesh University of Business and Technology, Dhaka, Bangladesh
e-mail: firoz@bubt.edu.bd

M. M. Kabir
e-mail: m97kabir2@gmail.com

A. Q. Ohi
e-mail: quwsarohi@bubt.edu.bd

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_7

plant pathologists often fail to diagnose specific diseases and lead to mistaken solutions. They may sometimes be misdiagnosed because of the wide variety of disease symptoms, and these symptoms look very close to each other. Genetic testing is preferable, but it is expensive and time-consuming.

Therefore, an efficient and highly accurate disease diagnosis and recognition method for plant diseases must be introduced. Several research approaches have proposed automated plant disease diagnosis methods that include pattern recognition [3, 4], machine learning [5], and deep learning [6]. Most of the techniques are plant-specific, and in some cases, they are disease-specific as well. Hence, multi-label classification is used in this work. Multi-label classifications can output several classes at the same time, from the same input image.

With the advancements of machine learning, computer vision applications have achieved enormous success. Success leads to implementing novel approaches and models, which now form a new class, known as deep learning (DL). DL techniques have been introduced in the agricultural domain, and they have gained massive popularity due to robustness. Researchers have invented convolutional neural networks (CNNs) that solve the pattern recognition problem associated with images. The transfer learning strategy has further enhanced the evolution of deep CNN-based architectures. In a transfer learning strategy, the CNN model is initially trained on a comparatively large dataset. The trained model is referred to as a pre-trained model. The pre-trained model further recognizes similar image patterns from the same or different domain of datasets. Transfer learning strategy often helps to avoid overfitting of DL architectures on small datasets. Furthermore, it also reduces the training iterations of DL architectures on the other datasets.

The overall contribution of the chapter can be summarized as,

- We investigate a multi-label CNN classifier that can identify multiple plants and the related plant diseases. This is the first research endeavor that consolidates numerous plants' diagnosis to the best of our knowledge.
- We combine six different plants, including Potato, Tomato, Corn, Rice, Grape, and Apples for our experiment. The experimental dataset contained a total of 28 diseases of the six plants.
- We experimented with six popular image recognition baseline strategies that include DenseNet, Inception, MobileNet, ResNet, VGG, and Xception. Different implementations of these popular baselines are also included, resulting in a total of 15 baseline implementations.

The rest of this chapter is organized as follows: Sect. 2 presents the related work. Section 3 introduces the dataset. In Sect. 4, the overall architecture of deep CNN is described. Section 5 contains the model's evaluation and compares the results of the architectures. Finally, Sect. 6 concludes the chapter.

2 Related Work

With advancements in computer vision, progress has been achieved in the identification and diagnosis of plant diseases. Numerous diagnosis and identification techniques are proposed by the following image segmentation procedures, feature extraction, and pattern recognition. Before the evolution of deep learning, the popular classification approaches that were used for disease detection in plants include random forest [7], artificial neural network (ANN) [8], k-nearest neighbor (KNN) [9], and support vector machine (SVM) [5]. Recognition methods using the procedures mentioned before improved plant disorder diagnosis. However, these approaches depend on the extraction and selection of visible disease features. Recently, several works on automated plant disease diagnosis and identification have been developed using deep learning techniques.

Kawasaki et al. [6] proposed CNN architectures to recognize cucumber leaf disease and obtained 94.9% accuracy. CNN is the most useful classifier for image recognition in both small and large-scale datasets. It has shown excellent performance in image processing and classification [10]. Mohanty et al. [11] trained a deep learning model for recognizing 14 crop species and 26 crop diseases with 99.35% accuracy using GoogleNet and AlexNet architecture. CNN can perform both feature extraction and image classification. Srdjan et al. [12] proposed a plant disease recognition approach to classify healthy leaves and 13 different diseases based on CNNs. The results demonstrate that robust computing infrastructure makes CNN a suitable candidate for disease recognition.

However, some defects and difficulties include collecting a large labeled dataset, which is challenging. Although CNN gives much better accuracy due to large datasets' unavailability, transfer learning approaches have been introduced in plant disease classifications. Transfer learning consists of a pre-trained network where only the last classification levels' parameters need to be inferred to obtain the classification results [13].

To date, no multi-label approaches have been introduced to explore multiple plant disease identification. This work aims to construct a robust multi-label transfer learning approach that will identify the plant and its disease with very low computational complexity and higher accuracy.

In this chapter, we study transfer learning for deep CNNs for multi-label plant disease identification and diagnosis techniques. Transfer learning utilizes knowledge from source models to improve learning in the objective task. Transfer learning reduces both training iterations and data required to achieve better results. Furthermore, due to the knowledge transform often transfer learning strategies perform better generalization.

3 Dataset

Six publicly available plant disease datasets are used during the experiments. The datasets were gathered from Kaggle [14]. The dataset contains leaf images of 6 different plants where each plant has 4–9 related diseases. Table 1 shows how the database is distributed in terms of plant species and diseases. Additionally, the number of images used in each class is shown in Table 1. Figure 1 shows the infected leaves of apple, namely, Apple scab, Black rot, and Cedar apple rust, with a healthy leaf image. Also, in Fig. 1, disorder images of corn are shown: Cercospora leaf spot, Common rust, and Northern leaf blight. This figure shows the Black measles, Black rot, and Leaf blight disease images of grape leaves. Early blight and Late blight are disordered potatoes, as shown in the figure. Three diseases of tomatoes, including Early blight, Late blight, Spider mites, are shown in Fig. 1.

Table 1 This table reports the plants and types of infection that our collected dataset contains

Plant name	Condition	Samples
Tomato	Healthy	1955
	Early blight	1955
	Late blight	1955
	Leaf mold	1955
	Septoria leaf spot	1955
	Spider mites	1955
	Target spot	1955
	Tomato mosaic virus	1955
	Yellow leaf curl virus	1955
Potato	Healthy	152
	Early blight	152
	Late blight	152
Corn	Healthy	2052
	Cercospora leaf spot	2052
	Common rust	2052
	Northern leaf blight	2052
Rice	Healthy	1046
	Brown spot	1046
	Hispa	1046
	Leaf blast	1046
Apple	Healthy	2200
	Apple scab	2200
	Black rot	2200
	Cedar apple rust	2200
Grape	Black measles	2115
	Black rot	2115
	Leaf blight	2115
	Healthy	2115

The count of each type of infection is represented in the 'Samples' column

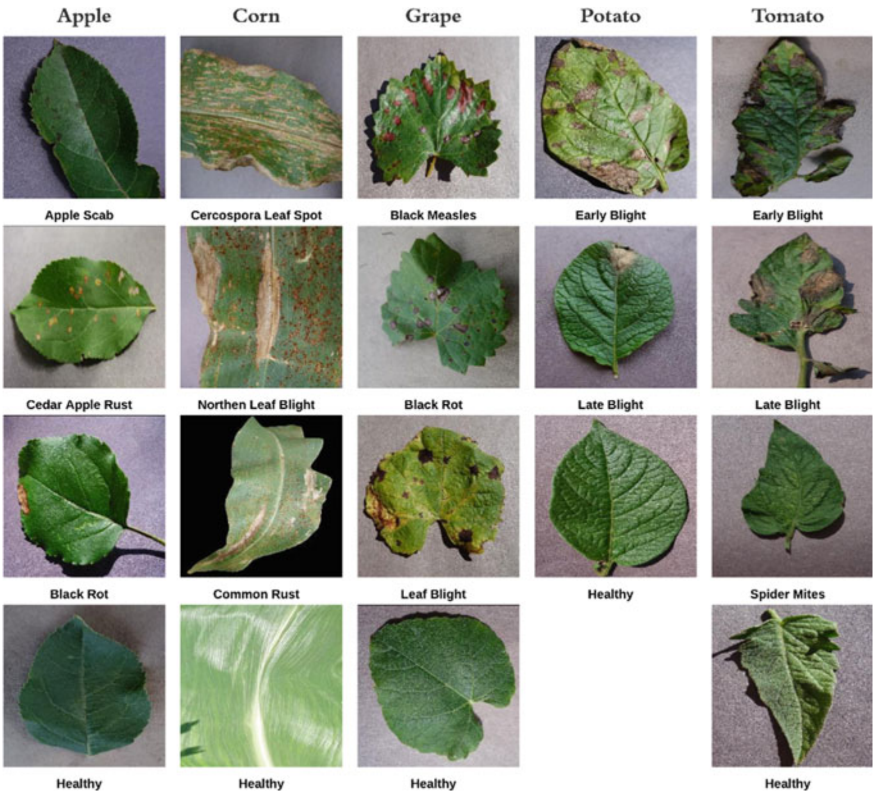


Fig. 1 This figure illustrates five plants and the types of diseases that each plant contains. Each row presents a set of leaf images caused by disease conditions for a specific plant

4 Methodology

Different CNN architectures are implemented and benchmarked to perform multi-label classification. In the following sub-sections, the general process and layers of various CNN architectures are briefed.

4.1 Image Pre-processing

As CNN architectures require input images to be of the same shape, each image data is reshaped into 120 by 120 pixels. Data normalization ensures that each input parameter has an analogous data distribution and results in faster convergence of the CNN. Therefore, each channel of the reshaped leaf images is normalized as,

$$\text{Normalize}(D) = \begin{bmatrix} d_{11} & \cdots & d_{1m} \\ \vdots & \ddots & \vdots \\ d_{n1} & \cdots & d_{nm} \end{bmatrix} / 255 \quad (1)$$

where D is the single-channel leaf image matrix, n is the number of rows, and m is the number of columns of the leaf image matrix.

4.2 Baseline Architecture

The concept of computer vision is to enable machines to understand the world as humans do. CNN is a deep learning technique that takes input, adds weights and biases, and classifies images. Efforts have been made to explain the methodologies of CNN architectures [15].

In this book chapter, we focus on implementing the network architecture for robust plant disease diagnosis. Figure 2 shows the CNN architecture, with the input layer (the raw image), convolutional layers, dense layer, and an output layer.

Input layer: The inputs of the CNN architecture is the raw leaf images of different plants. Leaf images with different widths and heights are resized into the shape of $120 \times 120 \times 3$ before being given to the CNN architecture.

Convolutional layers: Multiple convolution layers follow the input layer of the model. It is a mathematical operation that takes two inputs, an image matrix and a filter (or kernel), and conserves the relationship between pixels by learning image features. In every layer, there is a bank of m_1 kernels. Each layer identifies particular features at every location on the input. The output Y_i^l of layer l consists of the m_3^l feature of size $m_1^l \times m_2^l$. The *ith* feature map denoted Y_i^l, is computed as,

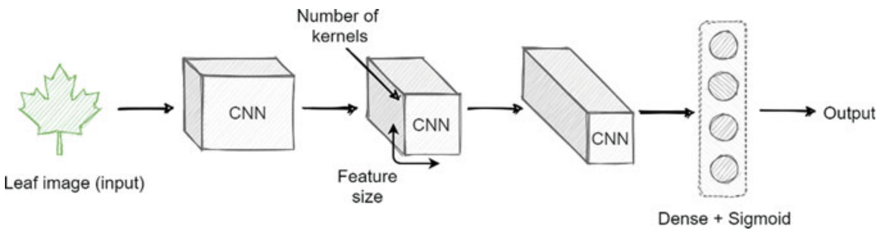


Fig. 2 The neural network architecture of plant diagnosis from leaf images. Each of the cubes represents an output of the convolution. The height and width are the gained information, and each cube's depth is equal to the number of kernels. Each convolution is followed by batch normalization and an activation layer. After the final convolution, it is converted into a linear set of nodes. Each node flows values through a sigmoid activation function

$$Y_i^{(l)} = B_i^{(l)} + \sum_{j=1}^{m_{\mathrm{i}}^{(l-1)}} K_{i,j}^{(l)} \times Y_j^{(l-1)} \tag{2}$$

where $B_i^{(l)}$ is a bias matrix and $K_{i,j}^{(l)}$ is the filter of size. Using convolution reduces the computational parameter. A fully connected layer would contain at least the same number of trainable parameters the same as the number of pixels in an input image. Such a large number of parameters often result in overfitting. On the contrary, convolution does not increase the number of parameters. Instead, it only searches for a specific feature matrix K that is learned through backpropagation. Recent deep neural network (DNN) architectures limit the height and width of the matrix K to a maximum value of 5. Hence the trainable parameters are balanced, and therefore convolution is used in the first layer of the architecture.

Batch normalization: Batch normalization changes inter-layer outputs into a standard format. Batch normalization re-calibrates each of the data values based on the mean and variance for a specific data batch. Batch normalization increases the stability of DNN architectures and often leads to faster convergence. The general formula of normalization is defined as follows,

$$x_i'^{(k)} = \frac{x_i^{(k)} - \mu_B^{(k)}}{\sqrt{\sigma_B^{(k)^2}}} \tag{3}$$

where $x_i'^{(k)}$ is the normalized value of the *kth* hidden unit. $\mu_B^{(k)}$ is the mean value, and $\sigma_B^{(k)^2}$ is the variance of the *kth* hidden unit. B defines the data of a particular batch.

ReLU: ReLU function is used in every convolution for a simple calculation that returns the value provided as input directly. The function returns zero if it receives any non-positive input, but for any positive value of x, it returns that value. So it can be written as,

$$\text{ReLU} = \max(0, x) \tag{4}$$

The non-linearity of ReLU causes DL architectures to detect the optimal position properly.

Dense layer: The dense layer, also called a fully connected layer, works on a flattened input where every input is connected to all neurons. Assigning a dense layer is a simple way of sensing nonlinear combinations of the high-level features extracted using previous CNN layers. A single dense layer is used in the architecture to classify the disorder. Mathematically, a dense layer can be represented as,

$$d(x) = \text{Activation}\big(w^T x + b\big) \tag{5}$$

Here, $w = [w_1, w_2, \ldots, w_n]^T$ represents the weight vector of the dense layer, and b represents the bias value of the dense layer. The activation function is a nonlinear function that defines the final output for a given input. A dense layer is used in the late part of the proposed deep neural network architecture to classify the plant species and type of the diseases. The sigmoid function is used as an activation function in dense layer.

Sigmoid: Sigmoid activation is used for multi-label classification of plant species identification and detecting disorder. It takes a real value as input and outputs another value [0, 1]. In our architecture, it defines plant species identification and disorder diagnosis. A sigmoid function can be represented as follows,

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{6}$$

Here, x represents the input of the sigmoid function. The sigmoid function is used as the final activation function represented in Eq. 6. Every node of the dense layer contains a sigmoid function for classifying the plant and disease.

4.3 Loss Function

To calculate the loss, we have used binary cross-entropy in the architectures. The binary cross-entropy loss function calculates the loss of an input that is stated below,

$$L_{\mathrm{ce}}(y, o) = -\sum_{l} (y_l \log o_l) - (1 - y_l) \times \log(1 - o_l) \tag{7}$$

Here, y and o are the target label and the output of the model, respectively. y_l is the target for label l, and o_l is the prediction for label l.

5 Evaluation

In this section, firstly, the evaluation metrics are defined. Later, the empirical setup is explained. Finally, we present the evaluation with a detailed analysis.

5.1 Evaluation Metric

We have used accuracy, precision, and recall evaluation metrics based on the confusion matrix. A confusion matrix summarizes prediction results that measure the performance for machine learning, deep learning classification problems that contain four measures: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). We evaluate the performance of the architecture by these measurements.

Precision: Precision defines all the positive classes the model predicted correctly; how many are actually positive. To obtain the value of precision, the total numbers of correctly classified positive examples are divided by the total number of predicted positive examples. The equation can be stated as,

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{8}$$

Recall: It defines how much the model predicted correctly among all positive classes. A recall is the ratio of the total number of correctly classified positive examples divided by the total number of positive examples. The equation can be stated as,

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{9}$$

F1-score: F1-score gives an overall estimation of the precision and recall of a test subject. It is the harmonic mean of the precision and recall of a test subject. Formally, F1-score can be defined as,

$$F1\_\text{score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{10}$$

5.2 Experimental Setup

Python is used to collect data, pre-processing, experiments, and evaluations of the model [16]. The neural network architecture is implemented in *Keras* [17]. *Numpy* [18] is used to perform basic mathematical operations. Also, we have used *Keras* to implement deep learning models. *TensorFlow* [19] is used to generate the GPU execution of neural networks. For as a pre-training, we used *ImageNet* weights for each of the models. The input shape of the leaf images is $120 \times 120 \times 3$. The dataset has been split into train, test, and validation sets with a percentage of 50%, 25%, and 25%, respectively. The Keras implemented Adam [20] optimization function is used to train the model for all the datasets with a learning rate of 0.001. A batch size of 128 is used for model training.

5.3 *Experiments and Comparisons*

For proper evaluation, each model was pre-trained on the ImageNet dataset. Each of the results presented in this chapter is presented as a mean of four runs. Each model is trained with a limit of 100 epochs. However, the training is halted if the loss on the validation dataset does not improve for ten epochs.

Figure 3 represents the F1-score on the validation dataset of the different models. From the presented graphs, it can be indicated that almost all of the architectures generate better F1-scores. However, due to overfitting in the training dataset, ResNet [21] architectures and VGG architectures hardly reach a minimum F1-score of 0.9. On the contrary, MobileNet architecture performs better compared to the trainable parameters available in the architecture. DenseNet, Xception, and Inception architectures avoid overfitting due to the skip connections. Among the various architectures, DenseNet169 and Xception reach the highest F1-score of 0.95.

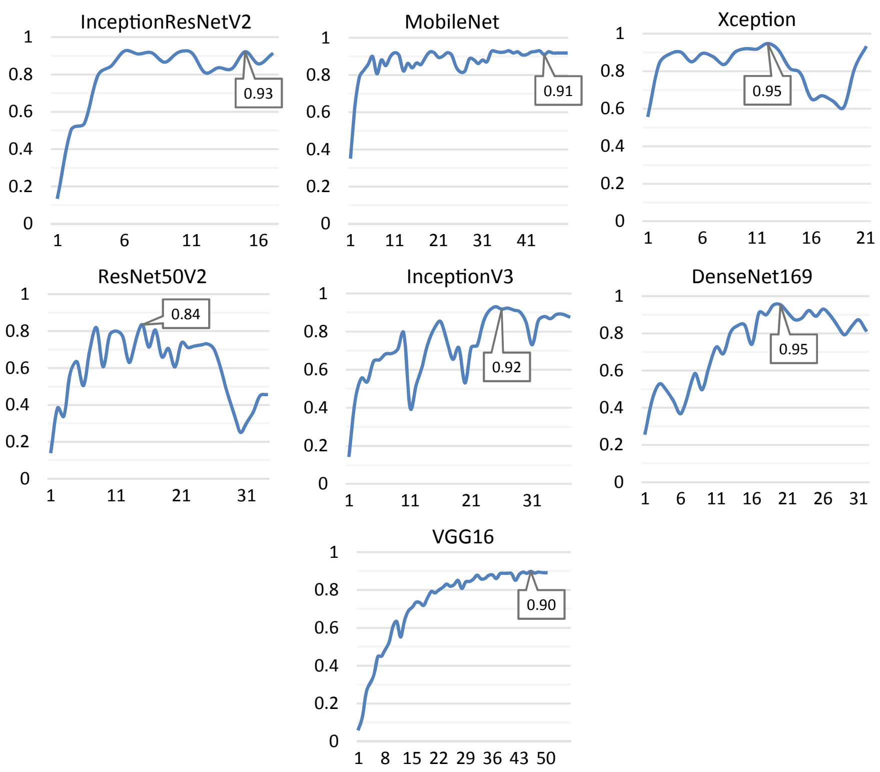


Fig. 3 For each CNN architecture, a separate graph is represented corresponding to the F1-score calculated on the validation dataset. Each graph's horizontal axis represents the epochs, and the vertical axis represents the F1-score in the scale of [0, 1]. The data illustrated are the mean of four runs

Table 2 This table represents the precision, recall, and F1-score of different CNN architectures measured on the validation dataset

Model	Parameters (million)	Precision	Recall	F1-score
DenseNet169 [22]	13	97.87	96.85	97.36
InceptionV3 [23]	23	95.9	95.19	95.55
InceptionResNetV2 [24]	54	96.92	93.95	95.41
MobileNet [25]	3	95.85	95.75	95.8
ResNet50V2 [26]	24	96.66	95.15	95.9
VGG16 [27]	137	92.53	89.62	91.05
Xception [28]	21	**97.88**	**96.9**	**97.38**

Furthermore, the number of trainable parameters of each CNN implementation is also reported. The highest score is marked in bold

The scores on the test data are reported in Table 2. The scores of each model are generated for the best weight found on the validation dataset. Xception performs better w.r.t. to the precision and recall on the test dataset. On the contrary, DenseNet169 slightly falls off from the maximum result. From the overall observation, it can be identified that Xception and DenseNet architectures perform better in the multi-label classification of the plant disease.

DenseNet architectures contain massive parallel skip connections. Skip connections refer to the process of forwarding the output of a node to skip some of the subsequent layers. DenseNet is designed based on the concept that convolution networks can perform better if they contain shorter linkage with the input layer and output layer. This method often helps architectures avoid overfitting and finding optimal combinations of neural activations for numerous input patterns.

On the contrary, Xception architecture is a combination of VGG and Inception architecture methodology. Furthermore, instead of general convolutions, Xception architectures perform spatial convolutions. Spatial convolutions work parallelly on multiple filters and tend to recognize texture features better.

The overall benchmarks and evaluation illustrate the recent improvements in the DL architectures in computer vision. The result states that the recently investigated architectures mostly perform better in image recognition tasks. Also, the chapter validates the usefulness of skip connections, spatial convolutions of the DNN architectures.

6 Conclusion

This chapter implements and tests a multi-plant diagnosis method validated based on different image classifier baselines. We practiced a transfer learning scheme to train and test our approach precisely. Further, we evaluate the architecture on a dataset consisting of 6 different plants and 28 different diseases. We observe that separable convolution and skip connections can massively improve multi-label plant disease detection performance. As no dataset is available for multi-label disease classification, we will consider adding more diverse plants and diseases to evaluate our methods in future work. We strongly believe that this research work's contribution will be regarded as complementary to the present work, paving the way for significant research on transfer learning approaches for plant and disease identification.

References

1. Strange, R.N., Scott, P.R.: Plant disease: a threat to global food security. Ann. Rev. phytopathol. **43**, (2005)
2. Ebrahimi, M.A., Khoshtaghaza, M.H., Minaei, S., Jamshidi, B.: Vision-based pest detection based on svm classification method. Comput. Electron. Agric. **137**, 52–58 (2017)
3. Asfarian, A., Herdiyeni, Y., Rauf, A., Mutaqin, K.H.: A computer vision for rice disease identification to support integrated pest management. Crop Prot **61**, 103–104 (2014)
4. Phadikar, S., Sil, J.: Rice disease identification using pattern recognition techniques. In 2008 11th International Conference on Computer and Information Technology, pp. 420–423. IEEE, (2008)
5. Deepa, S., Umarani, R.: Steganalysis on images using svm with selected hybrid features of gini index feature selection algorithm. Int. J. Adv. Res. Comput. Sci. **8**(5), (2017)
6. Kawasaki, Y., Uga, H., Kagiwada, S., Iyatomi, H.: Basic study of automated diagnosis of viral plant diseases using convolutional neural networks. In: International Symposium on Visual Computing, (pp. 638–645). Springer, (2015)
7. Kodovsky, J., Fridrich, J., Holub, V.: Ensemble classifiers for steganalysis of digital media. IEEE Trans. Inf. Forensics Secur. **7**(2), 432–444 (2011)
8. Sheikhan, M., Pezhmanpour, M., Moin, M.S.: Improved contourlet-based steganalysis using binary particle swarm optimization and radial basis neural networks. Neural Comput. Appl. **21**(7), 1717–1728 (2012)
9. Guettari, N., Capelle-Laizé, A.S., Carré, P.: Blind image steganalysis based on evidential k-nearest neighbors. In 2016 IEEE International Conference on Image Processing (ICIP), (pp. 2742–2746). IEEE (2016)
10. Kamilaris, A., Prenafeta-Boldú, F.X.: Deep learning in agriculture: a survey. Comput. Electron. Agric. **147**, 70–90 (2018)
11. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. Frontiers Plant Sci. **7**, 1419 (2016)
12. Sladojevic, S., Arsenovic, M., Anderla, A., Culibrk, D, Stefanovic, D.: Deep neural networks based recognition of plant diseases by leaf image classification. Comput. Intell. Neurosci. 2016 (2016)
13. Chen, J, Chen, J, Zhang, D., Sun, Y, Nanehkaran, Y.A.: Using deep transfer learning for image-based plant disease identification. Comput. Electron. Agric. **173**, 105393 (2020)
14. *Kaggle*, Accessed August 10, 2020. URL: www.kaggle.com
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In Advances in Neural Information Processing Systems, (pp. 1097–1105) (2012)
16. Van Rossum, G., Drake, F.L.: Python 3 Reference Manual. CreateSpace, Scotts Valley, CA (2009)
17. Chollet, F. et al.: Keras. https://keras.io, (2015)
18. Van Der Walt, S., Colbert, S.C., Varoquaux, G.: The numpy array: a structure for efficient numerical computation (2011)

19. Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M. et al.: Tensorflow: a system for large-scale machine learning. In 12th fUSENIXg symposium on operating systems design and implementation (fOSDIg 16), (pp. 265–283), (2016)
20. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980, (2014)
21. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (pp. 770–778), (2016)
22. Huang, G. et al.: "Densely connected convolutional networks." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (2017)
23. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision. *In* Proceedings of the IEEE conference on computer vision and pattern recognition. (pp. 2818–2826), (2016)
24. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.: Inception-v4, inception-resnet and the impact of residual connections on learning. arXiv preprint arXiv:1602.07261, (2016)
25. Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H.: Mobilenets: efficientconvolutional neural networks for mobile vision applications. arXiv preprintarXiv:1704.04861, (2017)
26. He, K, et al.: Identity mappings in deep residual networks. European Conference on Computer Vision. Springer, Cham (2016)
27. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, (2014)
28. Chollet, F.: Xception: Deep learning with depthwise separable convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, (pp. 1251–1258), (2017)

A Deep Learning-Based Approach for Potato Disease Classification

Md. Zahid Hasan, Nusrat Zahan, Nahid Zeba, Amina Khatun, and Mohammad Reduanul Haque

1 Introduction

Potato is an edible tuber of the Solanaceae family of the nurtured plant, scientifically known as Solanum tuberosum. It is the third most important food in the world after cereals and rice. It is also one of the main food crops in Bangladesh and the demand is also growing day by day as it is a rich source of starch, flour, alcohol, dextrin, fodder, and has high carbohydrate content. It also contains vitamin C, vitamin B, potassium, phosphorus, and iron. But potato production is threatened due to different diseases that bring about significant misfortunes along with causing a decline in the quality as well as raise the cost of potatoes [1]. According to a report, about 2–9% losses of tuber occurred and approximately 0.187 million tons of potatoes are damaged in Bangladesh because of different potato tuber diseases [2]. An early disease identification framework can help to reduce such cases. Furthermore, it can additionally control the spread of diseases [3].

Md. Z. Hasan (✉) · N. Zahan · N. Zeba · M. R. Haque
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh
e-mail: zahid.cse@diu.edu.bd

N. Zahan
e-mail: nusrat15-9716@diu.edu.bd

N. Zeba
e-mail: nahid25-863@diu.edu.bd

M. R. Haque
e-mail: reduanulhaque@gmail.com

A. Khatun
Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh
e-mail: amina_bashar@yahoo.com

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_8

Recently, computer vision and deep learning methods especially deep CNN give better accuracy in object detection and image recognition, also different computer vision methods have been altered from statistical methods into deep learning methods. It also showed a promising result in potato tuber disease identification problems. Basically, in this approach features are learned habitually from given data where traditional machine learning needs the training to learn features [4, 5].

As mentioned above, deep CNN has the ability of self-learning like it can extract features along with classifying images from input data, so in this work deep CNNs (i.e. AlexNet, GoogLeNet, and ResNet) have been effectively applied to identify different types of potato tuber diseases. Moreover, this study aims to find out fresh potatoes as well as improve the management and prevent the spread of diseases through deep learning approaches.

The rest of the chapter is organized as follows: related works are presented in Sect. 2. The methodology is described in Sect. 3. Section 4 presents the experimental results and finally, the chapter is concluded in Sect. 5.

2 Related Works

In agricultural applications like detecting crop diseases and so on, many methods have been studied and summarized in recent times. Besides, several methods have been applied and suggested for identifying potato tuber diseases by the researchers.

Utilizing the visual symptoms of potato tuber, Oppenheim et al. [6] proposed an advanced machine vision along with a deep learning approach for classifying potato tubers. The work used only five classes and among them, four classes were disease (black scurf, silver scurf, common scab, black dot) infected and the rest was healthy potato class. Their classification algorithm got an accuracy of 83–96% after testing 90% data.

Early blight and late blight are the two most infected diseases of potatoes that decrease production on a large extend. In this regard, Agarwal et al. proposed an approach based on deep learning for identifying these two diseases through examining different potato leaf's visual clarification. Their experiment considered some opposing circumstances like variable backgrounds, fluctuating image sizes, three-dimensional differentiation, a high-frequency disparity of grades of radiance, and a real scene image. Four convolution layers along with 32, 16, and 8 filters used in every layer of CNN is this work. Their suggested approach acquired an accuracy of 99.47% for training and 99.8% for testing [7]. Again, focusing on only early blight and late blight disease of potato, another work had done by Tiwari et al. for identifying infected potato leaves as it hampers the production and decreases the quality and quantity of potatoes. They applied a pre-trained VGG 19 model for extracting features from a significant dataset. They acquired a classification accuracy of 97.8% with the help of multiple classifiers amongst logistic regression overtook others through an extensive margin of classification [8]. Another approach of segmentation utilizing K means clustering made by Badar et al. focusing on various attributes of potato leaf image samples likely color, texture, area, etc. Propagation Neural Network algorithm is applied by them for properly detecting as well as classifying leaf disease of potato with 92% classification accuracy [9]. Furthermore, Islam et al. [10] completed research on the leaf image plant village dataset of potato-based on the concept of image segmentation. A multiclass support vector machine applied to that segmented image for detecting disease and got 95% classification accuracy.

From the above discussion, it is clear that very few works have been done before with insufficient datasets. Moreover, no methodology has been proposed before to classify as much as the disease of potato. The goal of this study is to use a machine vision and deep learning approach to classify and detect diseases as much as possible that is found in potato that brings an evolution in potato cultivation.

3 Materials and Methods

3.1 Dataset Description

In this experiment, about 1574 photographs of debased potatoes of various shapes, sizes as well as tones were obtained under different circumstances. The potato tubers were physically arranged by specialists as a standard methodology of factually evaluating the pace of different illnesses in seed tubers of potato before implanting them in the fields. This method is done yearly free of the flow research. The potatoes were defiled with twenty several infections along with huge visual manifestations on the tuber's peel showing in following Fig. 1. We have captured these images by utilizing different sorts of standard cameras and some of them are taken from the internet. Every potato picture is escorted with a sensorial explanation and then put away in a database which is then utilized for the turn of events and testing of the shading and shape calculations.

Figure 1 represents twenty types of potato tuber diseases such as black dot, black scurf, blackleg, brown rot, common scab, dickey, dry rot, gangrene, late blight, pink rot, powdery scab, rubbery rot, ring rot, skin spot, silver scurf, Tobacco Necrosis Virus (TNV), Tobacco Rattle Virus (TRV), violet root rot, watery wound rot, and wart disease. Besides, disease symptoms are represented in Table 1.

3.2 Deep Convolutional Neural Network

Deep learning which is known as a convolution neural network (CNN) has been widely used for image analysis for its effectiveness in recent years. This technique can directly find out high-level attributes through many hidden layers of network architecture from input image data. The convolution layer does the work of feature extraction from an input image using CNN. Rectified linear unit (ReLU), Pooling

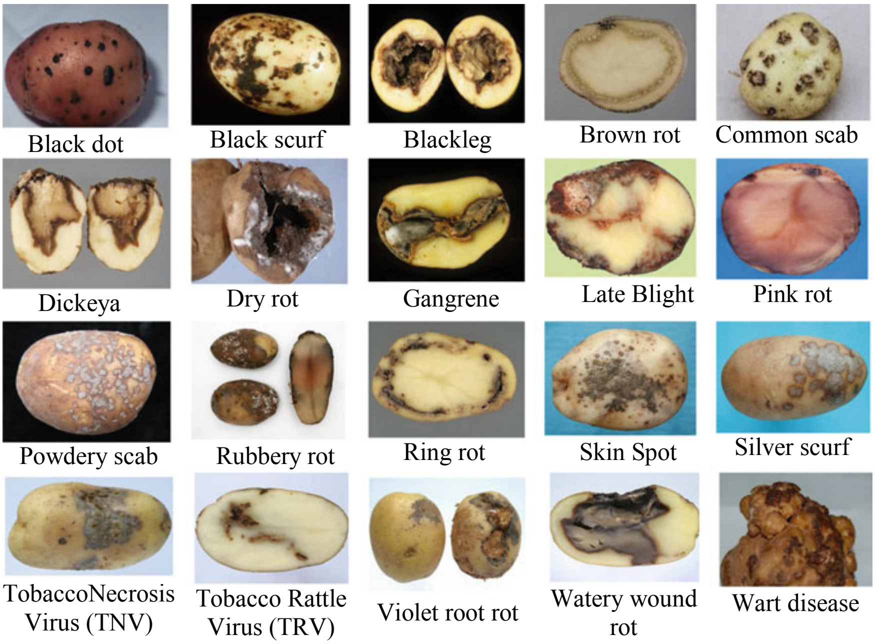


Fig. 1 Sample images of different potato diseases

or Downsampling layer, Fully Connected layers etc. are some of the other most important layers where the output of the hidden layer is served to the activation function of CNN. The network becomes more powerful by activation function along with adding the ability to learn complicated and complex features from data as well as representing non-linear complicated random functional mappings among input and output. Utilizing a nonlinear activation is a must to propagate a non-linear mapping from input to output. In this case, ReLU is used in the hidden layer, and on the other hand, SoftMax is used in output layers. The SoftMax activation function is considered as the most popular activation which mainly transforms score to the probability in the middle of 0–1. The following equation represents the probability distribution of SoftMax function:

$$\sigma(x_n) = \frac{e^{x_n}}{\sum_{n=1}^{i} e^{x_m}} \tag{1}$$

Here $m = 1, 2, 3 \ldots. i$ and $n = 1, 2, 3 \ldots. i$. Classification model performance measured by Cross-entropy loss or log loss considering probability value among 0 and 1 which is defined by following equation:

$$A(y, z) = E_y\left[-\log^z\right] = A(y) + D_{kl}(y||z) \tag{2}$$

Table 1 Symptoms indication of Potato disease dataset

Disease category	Symptoms details
Black dot	Appears with light brown colored to immaculate yet with dots
Black scurf	Ruddy brown color has shown up in Stems and stolon's
Black leg	Foliage turns in stunted, yellowish, and rot may get dry
Brown rot	A pale shaded clingy seepage may show up along with recoloring brown colored of the vascular ring
Common scab	dark brown, pithy patches that might be raised as well as "warty
Dickeya	Wet, black rot on the stem. Soft rots can be recognized as soft, watery plant tissue with small, wet stem lesions
Dry rot	Necrotic areas shaded from light to dark chocolate brown or black
Gangrene	Little round, dark depressions that may seem dark grey to brown
Late blight	Sores on leaves for the most part show up as sporadic formed dull spots which extend as the disease creates
Pink rot	Infected flesh may seem more white or light grey than the natural flesh tone but in an hour the cut surface could turn pink and later dark brown to black
Powdery scab	Appear as purplish-brown, round, slightly raised, and usually shallow lesions
Rubbery rot	Tubers might contain clusters of grey or white mycelia
Ring rot	Rots and the skin of the potato may crack
Skin spot	Tubers are covered by numerous raised ***spots*** surrounded by dark, sunken rings
Silver scurf	Silver scurf causes skin blemishes on potatoes that are tan to silver in appearance
Tobacco Necrosis Virus (TNV)	It is a rare tuber spot disease that appears as dark brown, dark sunken lesions and light brown
Tobacco Rattle Virus (TRV)	Appears as rust/brown arcs, concentric rings, or extensive browning of tuber tissue that later dry into corklike tissue
Violet root rot	Appears as dark, purple colored rot on the root's surface
Watery wound rot	Affected flesh can be dis-colored grey through to brown with a dark margin.
Wart disease	Tubers get de- shaped, warty

3.2.1 AlexNet

AlexNet is a deep convolutional neural network with 60 million parameters along with 650,000 neurons and is the champion of the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012 [11]. The following Fig. 2 represents the architecture of AlexNet used in this study. Five convolutional layers such as Conv1, Conv2, and so on complete AlexNet architecture. Pool1, pool2, pool5 and fc6, fc7,

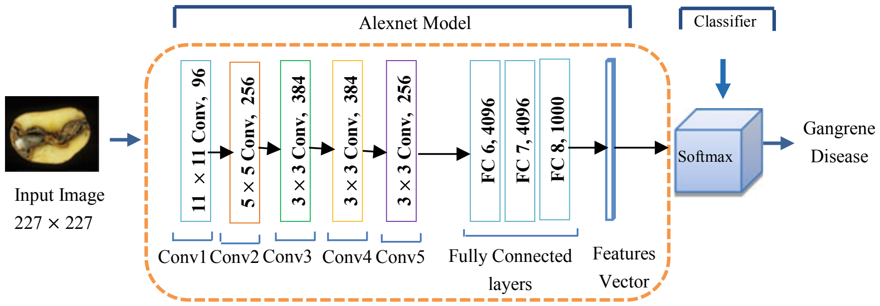


Fig. 2 The AlexNet architecture

fc8 are the max-pooling layer and three Fully Connected layers respectively along with a linear layer with SoftMax activation in output which follows some of the layers of this architecture.

Dropout such as drop6, drop7 is a regularization method used for reducing overfitting in the Fully Connected layer in this network architecture [12]. ReLU1, ReLU2, and so on are each of the first seven layers where the ReLU (Rectified Linear Unit) activation function is applied [13]. The size of the feature map of every layer is represented by the notation $m \times m \times n$ in every convolutional layer in Fig. 2. The first two Fully Connected layer number of neurons are considered as 4096. According to AlexNet requirements, the input pixel size must be shaped to 227 × 227. The contribution should contain no more than four levels of headings. The following Table 1 gives a summary of all heading levels.

3.2.2 GoogLeNet

GoogLeNet is none but inception v3 architecture with 6.8 million parameters and the winner of ILSVRC 2014 [14]. The following Fig. 3 represents the architecture of this network model.

A parallel amalgamation of 1 × 1, 3 × 3, and 5 × 5 convolutional layers with 3 × 3 max-pooling layers are used by this inception model. The task of reducing the spatial dimension as well as limiting the GoogLeNet size is completed by the 1 × 1 convolutional layer before 3 × 3 and 5 × 5 convolutional layer.

Total nine inception modules with two convolutional layers, four max-pooling layers, one average pooling layer as well as one Fully Connected layer along with a linear layer connected with SoftMax function in the output form the whole inception module which further stacks the complete GoogLeNet architecture. In this architecture, drop regularization is used in each convolutional layer of a Fully Connected layer by smearing the ReLU activation function.

A Fully Connected layer, a SoftMax layer, and a classification layer replace the last three layers of GoogLeNet in this work. The GoogLeNet requirement for the input pixel is 224 × 224.

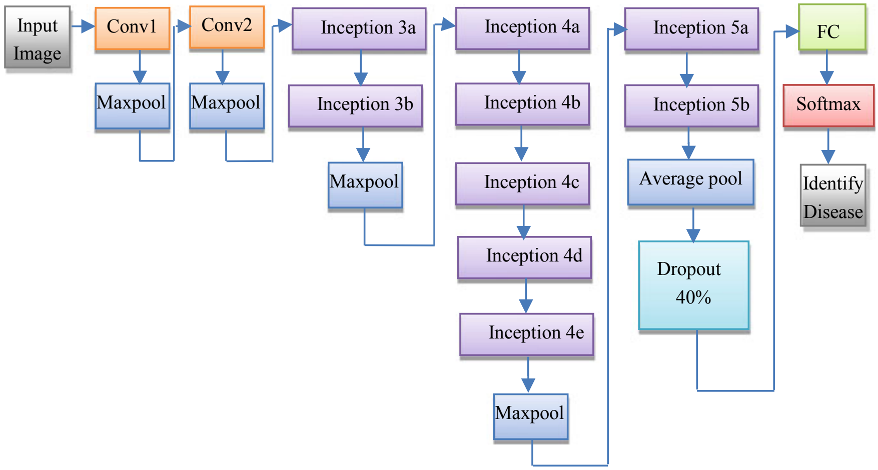


Fig. 3 The GoogLeNet architecture

3.2.3 ResNet-50

Since Convolutional neural networks have demonstrated to be viable in portrayal learning along with utilizing convolutional filters for extracting features as well as training the parameters via backpropagation, ResNet-50 [15], pre-trained on the ImageNet dataset and art convolutional neural network utilized by us. ResNet (see Fig. 4) gained first place in ILSVRC2015 and as well as COCO2015 classification challenge along with an error rate of 3.57% [15]. The following equations represent every unit of residual units which form ResNet architecture:

$$a_k = i(y_k) + F(y_k, Z_k) \tag{3}$$

$$y_{k+1} = f(a_k) \tag{4}$$

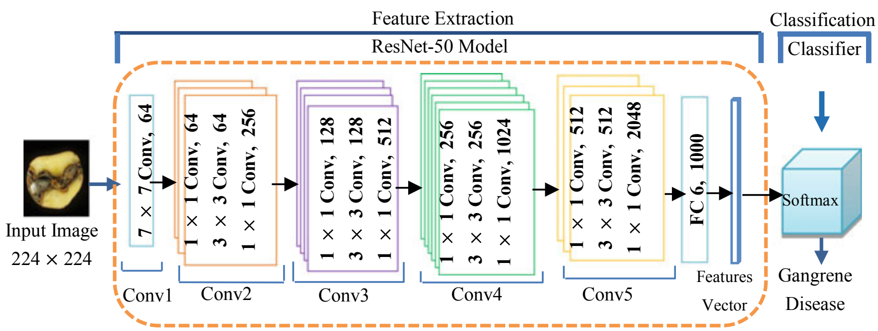


Fig. 4 ResNet-50 convolutional neural networks with SoftMax

Here, y_k and y_{k+1} represent the input and output of the kth unit where is a residual function.

According to the input requirements of the ResNet-50, images need to be resized into 224×224 to appropriately fitting into it. The attributes are acquired by expelling the last completely associated layer to acquire the 2048-dimensional component vector. These element vectors were gotten effectively without the utilization of much computational force. ResNet-50 can give a powerful component to most pictures.

4 Experimental Results and Discussions

For image classification, deep CNN is considered as a standard method because of its remarkable improvement over convolutional machine learning methods. The dataset consists of 7870 images. The dataset was divided into a training set and testing set in a ratio of 8:2 (see Table 2).

Table 2 Dataset of potato disease images

Diseases	Original	Expanded	Training	Test
Black dot	64	320	256	64
Black scurf	72	360	288	72
Blackleg	112	560	448	112
Brown rot	96	480	384	96
Common scab	45	225	180	45
Gangrene	93	465	372	93
Late Blight	55	275	220	55
Dickeya	131	655	524	131
Pink rot	87	435	348	87
Dry rot	162	810	648	162
Powdery scab	75	375	300	75
Rubbery rot	29	145	116	29
Ring rot	56	280	224	56
Skin spot	68	340	272	68
Silver scurf	109	545	436	109
Tobacco Rattle Virus	41	205	164	41
Tobacco Necrosis Virus	65	325	260	65
Violet root rot	50	250	200	50
Watery wound rot	72	360	288	72
Wart disease	92	460	368	92
Total	1574	7870	6296	1574

The data augmentation procedure was regulated by flipping the image, rotating it by 90° and 270°, adjusting the brightness, contrast, and saturation of the image.

4.1 Performance Matrix

The experiment of the chapter is combined with three pre-trained network AlexNet, GoogLeNet, and ResNet. Where ResNet obtained more accuracy rate in finding the potato diseases rather than other methods.

$$\text{Accuracy} = \frac{\text{true positive} + \text{true negative}}{\text{positive} + \text{negative}} \tag{5}$$

Here, true positive is an instance which is considered positive, a true negative is an instance which is considered as negative and the total numbers of samples are represented by the denominator.

$$\text{Precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \tag{6}$$

It is the ratio of correctly predicted positive observations of the total predicted observations.

$$\text{Recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \tag{7}$$

Recall or sensitivity is the ratio which predicts the positive observation correctly to all the observations in an actual class.

$$F1\ \text{score} = 2 \times \left(\frac{\text{recall} \times \text{precision}}{\text{recall} + \text{precision}}\right) \tag{8}$$

It is the weighted average of recall and precision and the result takes both positive and negative values.

For every potato disease number of images used which were collected from the internet and some were real environmental images. Each original image numbers were expanded by five times than original images and those expanded images were divided into 80% training and 20% testing dataset. Total training set 6296 and 1574 were the total number of test set images in the dataset.

To properly demonstrate the performance of the Google net, AlexNet and ResNet-50, twenty different variations of potato diseases of different confusion matrices are shown in Figs. 5, 6 and 7. From the figure, it is seen that the accuracy of x label represents the predicted label accuracy where y represents the true label accuracy. From the figure, it is also seen that most of the classes obtain a satisfactory identification result over 85% but the GoogLeNet shows some misclassification.

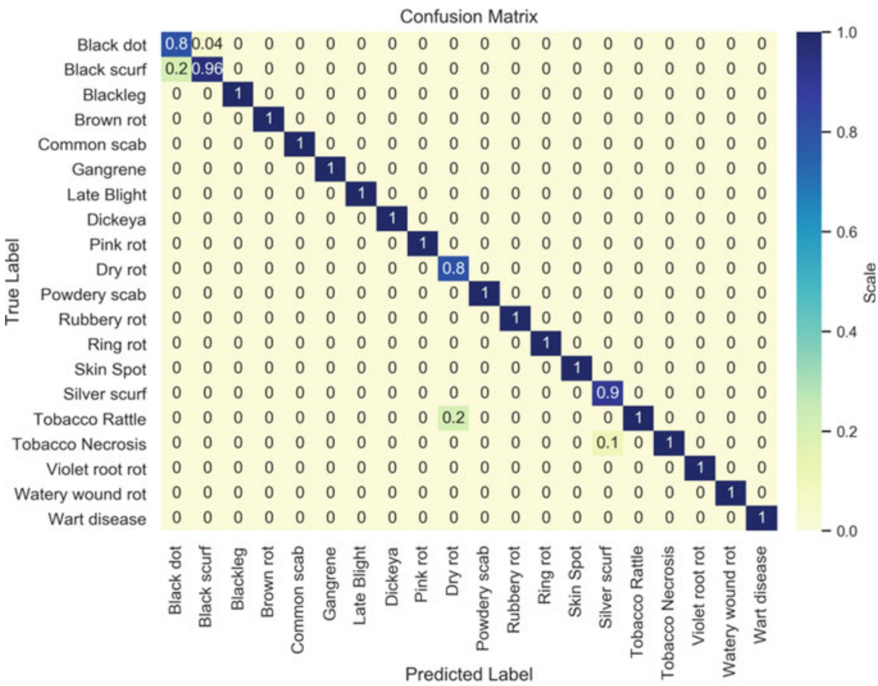


Fig. 5 Confusion Matrix for the GoogLeNet model of 20 different potato diseases

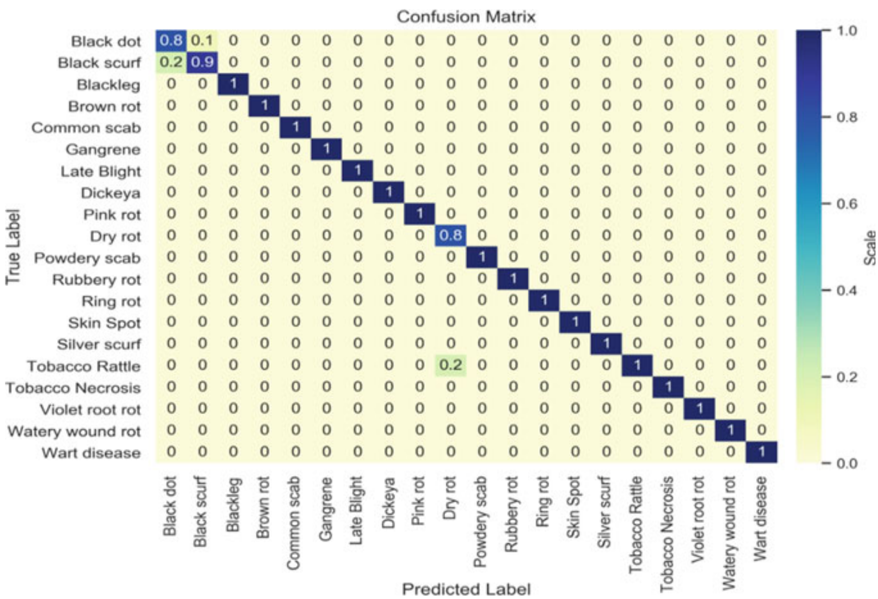


Fig. 6 Confusion Matrix for the AlexNet model of 20 different potato diseases

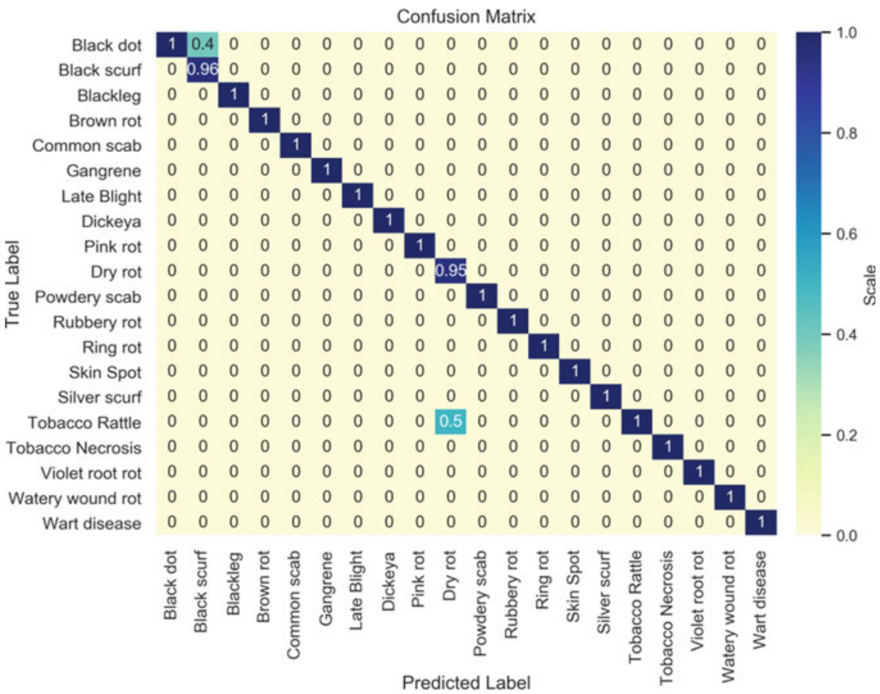


Fig. 7 Confusion Matrix for the ResNet-50 model of 20 different potato diseases

Table 3 Performance of AlexNet, GoogleNet, and ResNet-50 on Potato Tuber Datasets

Model	Accuracy (%)	Precision (%)	Recall (%)	F1 Score (%)
AlexNet	95.79	94.54	89.23	94.39
GoogLeNet	94.92	93.76	88.71	93.68
ResNet-50	**98.52**	**97.49**	**94.65**	**97.34**

The accuracy, precision, recall, and F1 score of three models for potato disease recognition are displayed in Table 3 where ResNet-50 obtains the highest performance having 98.52% accuracy, 97.49% precision, 94.65% recall, and 97.34% F1 score. On the other hand, the accuracy level of AlexNet and GoogLeNet were very close to each other and less than the ResNet-50 (see Fig. 8).

4.2 Number of Iterations

In this chapter, the training loss of ResNet-50 drops very frequently in earlier iterations and stable after 3000 iterations (see Fig. 9) and a total of 70,000 iterations were

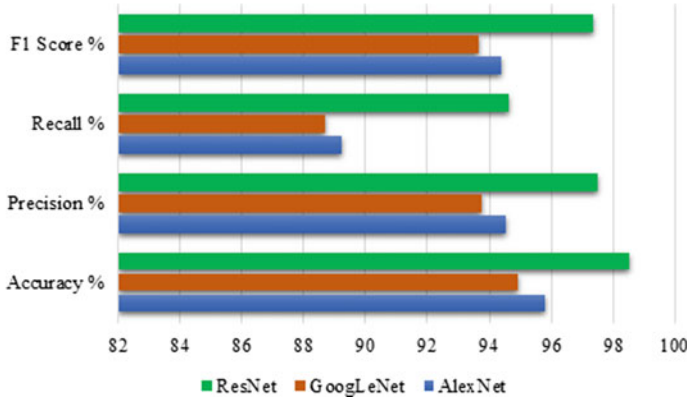
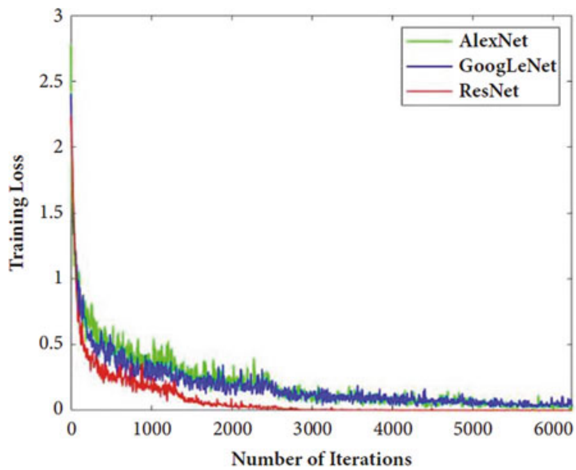


Fig. 8 Model recognition accuracy, precision, recall and F1 score

Fig. 9 Training loss of ResNet



done here. The performance of ResNet-50 was affected by iterations and the size of the batch was evaluated where the size of the batch was set to 16, 32, and 64. On the contrary, iterations were set to 2392, 4888, and 9870. The learning rate was set to 0.001 and dropped by 0.5 factors in every 2392 iterations so a medium number of iterations was very much functional here. The greater number of iterations and the number of batch sizes, the more time system takes for training.

The performance of previously trained CNN was compared through full training and fine-tuning structures and those were defined by the number of training layers.

After looking at the experimental results of three methods, it is examined that ResNet-50 gives much better performance and higher accuracy in identifying potato diseases from images than AlexNet and GoogLeNet. The whole experiment was implemented under Windows 10, using GPU NVIDIA GTX1080Ti with 11 GB video memory.

5 Conclusion

This chapter aims to identify diseases of potato using deep convolutional neural networks. About 1574 photographs of debased potatoes of various shapes, sizes as well as tones were used in the dataset for training purposes and expanded to 7870 images through the augmentation process, were divided into training and testing set. Almost 20 potato disease images were used in the dataset for learning techniques. Three deep learning methods ResNet-50, GoogLeNet, and AlexNet are used for detecting potato diseases, and by comparing the performance of these three models ResNet-50 obtained the highest result having accuracy 98.52%, precision 97.49%, recall 94.65%, and F1 score 97.34%. In the future, our focus will be broader by detecting more diseases of potato tubers. As potato is one of the main food crops in Bangladesh, so detecting diseases at an early stage is very important so that farmers can take necessary action to get rid of these diseases and imbalance demand and supplies in the market due to less productivity.

References

1. Taylor, R.J., Pasche, J.S., Gudmestad, N.C.: Susceptibility of eight potato cultivarsto tuber infection by phytophthora erythroseptica and pythium ultimum and its relationship to mefenoxam-mediated control of pink rot and leak. Ann. Appl. Biol. **152**(2), 189–199 (2008)
2. Anonymous Yearbook of Agricultural Statistics of Bangladesh. Bangladesh Bureau of Statistics. Ministry of planning, Govt. of the People's Republic of Bangladesh. pp. 47–53(2006)
3. Rich, A.E.: Potato Diseases. Elsevier Science. Academic Press, NewYork, USA (2013)
4. Huang, S., Fan, X., Sun, L., Shen, Y., Suo, X.: Research on classification method of maize seed defect based on machine vision. J. Sens. **2019**, 1–9 (2019)
5. Kiratiratanapruk, K., et al.: Development of paddy rice seed classification process using machine learning techniques for automatic grading machine. J. Sens. **2020**, 1–14 (2020)
6. Oppenheim, D., Shani, G.: Potato disease classification using convolution neural networks. Adv. Anim. Biosci. **8**(2), 244–249 (2017)
7. Agarwal, M., Sinha, A., Gupta, S.K., Mishra, D., Mishra, R.: Potato crop disease classification using convolutional neural network. In: Somani, A., Shekhawat, R., Mundra, A., Srivastava, S., Verma, V. (eds.) Smart Systems and IoT: innovations in Computing. Smart Innovation, Systems and Technologies, vol. 141. Springer, Singapore (2020)
8. Tiwari, Ashish, M., Gangwar, N., Sharma, A., Patel, S., Bhardwaj, S.: potato leaf diseases detection using deep learning. 4th International Conference on Intelligent Computing and Control Systems (ICICCS), pp. 461–466, Madurai, India (2020)
9. Athanikar, G., Badar, M.P.: Potato leaf diseases detection and classification system Mr. Int. J. Comput. Sci. Mobile Comput. **5**(2), 76–88 (2016)
10. Islam, M., Dinh, A., Wahid, K., Bhowmik. P.: Detection of potato diseases using image segmentation and multiclass support vector machine. 2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE), pp. 1–4, Windsor, ON, (2017)
11. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In Proceedings of the 26th Annual Conference on Neural Information Processing Systems (NIPS), pp. 1097–1105, Lake Tahoe, Nev, USA, December (2012)
12. Pawara, P., Okafor, E., Surinta., O., Schomaker, L., Wiering M.: comparing local descriptors and bags of visual words to deep convolutional neural networks for plant recognition.

In: Proceedings of the 6th International Conference on Pattern Recognition Applications and Methods, pp. 479–486, Porto, Portugal, February (2017)
13. Szegedy, C., Liu, W., Jia, Y. et al.: Going deeper with convolutions. in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1–9, IEEE, Boston, Mass, USA (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, pp. 770–778 (2016)

An In-Depth Analysis of Different Segmentation Techniques in Automated Local Fruit Disease Recognition

Md. Tarek Habib, Amina Khatun, Md. Abdul Aziz, Mohammad Shorif Uddin, and Farruk Ahmed

1 Introduction

Bangladesh, having an enormous population and fertile land enriched by heavy silt, is directly or indirectly dependent on agriculture. The agriculture sector performs remarkable feats to make sure the continual food security for this enormous population of Bangladesh [1]. It delivers, following the data published by the World Bank [2], a significant portion of the entire employment of Bangladesh, which quantifies to greater than 39%. So, it necessitates the agricultural products to be taken meticulous care to make quality sure. Disease-free production is regarded as one of the serious issues of quality. Machine vision-based, viz. automated local fruit disease recognition, which involves a machine vision application, can help a lot in this regard. One of the important steps in automated local fruit disease recognition is image segmentation, which is often used to partition an image of a diseased fruit into separate

Md. T. Habib (✉) · A. Khatun · M. S. Uddin
Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh
e-mail: md.tarekhabib@yahoo.com

A. Khatun
e-mail: amina_bashar@yahoo.com

M. S. Uddin
e-mail: shorifuddin@gmail.com

Md. T. Habib · Md. A. Aziz
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh
e-mail: aziz15-8396@diu.edu.bd

F. Ahmed
Department of Computer Science and Engineering, Independent University, Bangladesh, Dhaka, Bangladesh
e-mail: farruk60@gmail.com

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_9

regions including the defective part. It is a vital stride headed for analysis of contents and understanding of images. Thus, it severely affects the following steps, especially feature extraction. Quality feature extraction results in features of high distinguishing quality [3]. On the other hand, there is no stand-alone best technique for image segmentation. The preponderance of a specific image processing techniques depends on the problem domain, i.e., context. That is why it remains still difficult enough to assess whether one technique performs better than another in a given context, although there exist different on-hand techniques for image segmentation.

In this chapter, we conduct a profound comparative analysis of four on-hand image segmentation techniques, namely Otsu's method, K-means clustering, fuzzy c-means clustering, and region growing [4] in the context of automated local fruit disease recognition. Among various local fruits, we have selected guava, jackfruit, and papaya—the three common local fruits in Bangladesh, which are enormously yielded not only in farmlands but also home-side gardens by nonprofessional gardeners [5]. To measure the performance of different image segmentation techniques, two major types of evaluation, as discussed in [6] and [7], are used, such as subjective and objective. Subjective evaluation is the widely practiced way of measuring the performance of an image segmentation algorithm, where a person visually examines the results of different image segmentation techniques separately. For example, Zhang et al. [6], and Wang and Wang [8] have tried to use subjective evaluation of segmentation. However, it is lengthy, tedious, and even unfeasible in real-world applications as claimed in [6] and [9]. Moreover, different humans may provide inconsistent evaluations for the case of the segmentation, which are visually close. To overcome these limitations, an alternative way of measuring the performance of a segmentation technique has been used. It is called objective evaluation, where the performance of an image segmentation technique is evaluated automatically and rigorously. In the case of the objective evaluation, one or more performance metrics are used for performance evaluation. For example, Unnikrishnan et al., Hebert [10], Shi et al. [7], Kumar and Suhas [11], and Monteiro and Campilho [12] have proposed performance metrics for objective evaluation of segmentation. It is a matter of fact that some specific metrics are used for some specific machine vision or imaging applications. Owing to the dependence of feature extraction on image segmentation, six region-based metrics are used so that the quality of the segmentation techniques can be quantified and compared. These metrics are overlap measure, under segmentation measure, over segmentation measure, dice similarity measure, accuracy, and error rate. Almost all of these region-based metrics are proposed by Kumar and Suhas [11] and Monteiro and Campilho [12], and the rest is by us. The results obtained after calculating the six region-based metrics dictate the extraction of discriminatory features.

The upcoming portion of our chapter is arranged in the way as follows. Section 2 discusses the current situation for assessing the performances of different on-hand segmentation techniques in different contexts, especially the context of automated local fruit disease recognition. Section 3 delineates the complete research methodology of our work along with the discussion. Section 4 describes all diseases of all of the three local fruits as well as the six performance metrics used. Section 5 explains how the experiments are performed and the results obtained thereafter. At last, the summary and conclusion along with prospects are provided in Sect. 6.

2 Related Works

The problem of machine vision-based, i.e., automated fruit disease recognition, is broken down into two subproblems, viz. disease detection and disease classification. A good number of research works have been done on entire automated fruit disease recognition, but some works have been done on disease detection only leaving the classification undone. All these works vary not only across data size, techniques, and the number of diseases but also across the fruits themselves. Nonetheless, one thing is common to all of them. That is, not a single significant work has been done in quest of a segmentation technique among on-hand segmentation techniques, which outperforms all the techniques in terms of meaningful performance metrics, although some works on the comparison of different image segmentation techniques, e.g., [13], have been performed on different application domains. A sample experiment has been performed on a single standard test image, namely Lenna [14] using different image segmentation techniques.

Chopaade and Bhagyashri [15] performed a weak work based on image processing to perform only disease detection on the leaves of some fruits, namely banana, mango, and papaya, leaving the fruits themselves. Since they have carried out histogram-based segmentation only to detect disease of fruit leaves, no question for comparison arises rendering the work poor. Rozario, Rahman, and Uddin [16] have performed disease detection on some vegetables and fruits, namely apple, banana, potato, and tomato using image processing techniques. Although they have worked with some segmentation techniques to select the best one, the basis of selection is not rigorous. They have used only one measure rather than using a sufficient number of meaningful performance metrics. Wang et al. [17] have worked on orange skin disease detection using a machine vision approach, but they remained silent in case of segmentation. That means they have made use of a color histogram to form feature vector and linear support vector machine (SVM) to detect defective part(s) bypassing the step of segmentation. Samajpati and Degadwala [18] have come up with an amalgamated, i.e., hybrid approach for the recognition of the diseases of apples. They have professed fusion of features by amalgamation since two feature types, viz. texture-based and color-based features, have been used in their work. Three types of apple infections were pondered in the patchy work. The infested portion of a diseased apple image has merely been isolated by using the *K*-means clustering segmentation technique; nothing more has been done on segmentation. That means neither any other segmentation technique nor any segmentation performance metric has been used. Kumar and Suhas [11] have professed to use a machine vision-based method to recognize a variety of fruit diseases without the support of relevant information, which has rendered their work controversial. Although they have tried to investigate the performances of only two segmentation algorithms, viz. *K*-means clustering and fuzzy *c*-means clustering, the absence of many important details eclipses their attempt. Habib et al. [19] have introduced an approach based on machine vision to recognize papaya diseases. They have applied *K*-means clustering segmentation to separate the disease-affected portion for extracting the features. They have worked with a single segmentation technique. They have tried to assess the quality of segmentation with a single measure, but the use of a sufficient number of meaningful performance metrics on different on-hand segmentation techniques has been absent in their work. Three detached classifiers have been used to accomplish classification. Though the work is further continued by Habib et al. [20] applying six more illustrious classifiers, nothing more has been done on image segmentation. Habib et al. [21] have used a machine vision-based approach to recognize jackfruit diseases. They have applied *K*-means clustering segmentation to isolate the diseased portion for extracting the features. They have worked with a single segmentation technique. They have tried to assess the quality of segmentation with a single measure, but the use of a sufficient number of meaningful performance metrics on different on-hand segmentation algorithms has been absent in their work. Nine classifiers have been applied separately to accomplish the classification, but no in-depth investigation has been done on image segmentation. Majumder et al. [22] have used the same approach as the approach used by Habib et al. [20] to recognize carrot diseases. For extracting the features, like Habib et al. [20], they have solely applied *K*-means clustering segmentation to isolate the diseased portion. Only SVMs have been applied to accomplish the classification, but not even a little bit of investigation has been done on image segmentation.

Hosen et al. [23] have used an approach based on color segmentation to detect fruit defects. Their work has been confined to the only detection of the defective parts of the fruits, viz. banana, orange, and apple from 3D images. That is, they have dealt with 3D fruit images. They have also engaged a classifier to classify the specific disease from the captured images. Likewise, Rachmawati et al. [24] have done another work with 3D images in the domain of different fruits rather than fruit defects or fruit diseases. Although they have used seven different fruits, e.g., apple, banana, lemon, lime, orange, peach, and pear, they have used only one segmentation technique, i.e., *K*-means clustering taking the depth and color data into account. The performance of segmentation has been measured in terms of only a single metric, namely mean squared error (MSE). However, they have found an MSE of 0.54 using only color data and obtained an MSE of 0.28 combining color and depth data. But, working with 3D images is computationally expensive and complicated. Moreover, the potential overlap between images from different orientations renders the work more complicated.

3 Research Methodology

Our approach for analyzing different segmentation techniques on the background of automated local fruit disease recognition is delineated as a block diagram in Fig. 1. It commences an RGB color image of a diseased local fruit, i.e., guava or jackfruit or papaya. First, a conversion from this image happens into a preset size image by using bicubic interpolation [25]. Presuming that I is the intensity values, and I_x, I_y, and I_{xy} are the derivatives that can be experienced at the set of the quad corner points $\{(x, y) \mid x = y = 0, 1\}$ of the unit square, the interpolated intensity surface is expressed as:

$$s(x, y) = \sum_{m=0}^{3} \sum_{n=0}^{3} c_{mn} x^m y^n, \tag{1}$$

where c_{mn} is a coefficient.

Then the image contrast is enhanced by using the histogram equalization technique. By letting n be the count of pixels through the column, i.e., width, m be the count of pixels through the row, i.e., height, c_j be the count of pixels containing color intensity I_j, and l be the count of permissible color intensity levels in the image, and then mapping each of the pixels containing color intensity I_j into a corresponding pixel containing color intensity I'_j applying histogram equalization, the color-mapped image is found in the following way [26].

$$I'_j = T(I_j) = \frac{l-1}{mn} \sum_{i=0}^{j} c_i \tag{2}$$

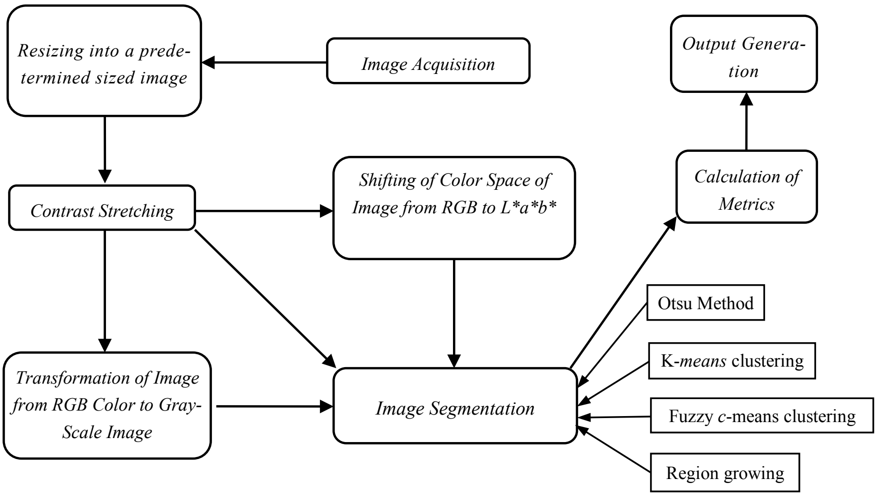


Fig. 1 Approach for analyzing different off-the-shelf segmentation techniques

where $j = 0, 1, \ldots, l - 1$.

Then this image goes through one of the following steps, as shown in Fig. 1, depending on the segmentation algorithm that is going to be used.

(1) Conversion of color space from RGB to $L^*a^*b^*$
(2) Conversion of color space from RGB to gray scale
(3) No conversion.

The image can go through the conversion from the color space of RGB to the gray scale so that the grayscale histogram can be formed. If R, G,, and B indicate the red, green, and blue color values of an RGB color pixel, respectively, and Y indicates the equivalent grayscale value of the same pixel, then Y can be computed in the following way as recommended in ITU-BT.709 [27].

$$Y = \frac{33 \times R + 56 \times G + 11 \times B}{100}. \tag{5}$$

The image can also go through the color space transformation from RGB to $L^*a^*b^*$ because of the claim of Burney and Tariq in [28] that the K-means clustering algorithm performs better in image segmentation in $L^*a^*b^*$ color space than that in RGB color space. Firstly, the color space conversion is conducted, as claimed in [29], from RGB to International Commission on Illumination (CIE) XYZ in the way as follows.

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} 3.240479 \times 10^{0} & -1.53715 \times 10^{0} & -4.98535 \times 10^{-1} \\ -9.69256 \times 10^{-1} & 1.875992 \times 10^{0} & 4.1556 \times 10^{-2} \\ 5.5648 \times 10^{-2} & -2.04043 \times 10^{-1} & 1.057311 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix}. \tag{4}$$

To convert color space from XYZ to $L^*a^*b^*$ color space, X_w, $Y_{w,}$ and Z_w are supposed as the tristimulus values of white—the reference color. If it is further supposed that

$$f(t) = \begin{cases} \sqrt[3]{t} & \text{if } t > 8.856 \times 10^{-3} \\ 7.787t + \frac{16}{116} & \text{if} \le 8.856 \times 10^{-3} \end{cases}, \tag{5}$$

we can equate L^*, a^*, and b^*, as described in [29], in the fashion as follows.

$$L^* = \begin{cases} 116\left(\frac{Y}{Y_w}\right)^{\frac{1}{3}} - 16 & \text{if} \frac{Y}{Y_w} > 8.856 \times 10^{-3} \\ 903.3\frac{Y}{Y_w} & \text{if} \frac{Y}{Y_w} \le 8.856 \times 10^{-3} \end{cases}. \tag{6}$$

$$a^* = 500 \times \left\{ f\left(\frac{X}{X_w}\right) - f\left(\frac{Y}{Y_w}\right) \right\}. \tag{7}$$

$$b^* = 200 \times \left\{ f\left(\frac{Y}{Y_w}\right) - f\left(\frac{Z}{Z_w}\right) \right\}. \tag{8}$$

Then the image is divided into several regions by applying one of the four prominent image segmentation techniques—region growing, K-means clustering, fuzzy c-means clustering, and Otsu's method. Thus, diseased parts are isolated from disease-free parts. A vector containing discriminatory features can be constructed from the attacked portion. It has already been mentioned that quality feature extraction hinges on the quality segmentation. So, for assessing the quality of the four segmentation techniques, six region-based performance metrics are calculated, which are being described in detail in the next section, i.e., Sect. 4.1.

4 Local Fruit Diseases and Performance Metrics

4.1 Disease Description

Visual investigation plays a very important role in our research work because it serves to appropriately comprehend the imperfections brought about by local fruit diseases and select candidate segmentation techniques to come up with the most appropriate one. We have worked with four diseases, which often happen everywhere throughout the country, for each of the three common local fruits of Bangladesh, i.e., guava, papaya, and mango. That means we have dealt with twelve diseases in total. The outbreaks of all of these twelve diseases are pictorially exhibited in Fig. 2.

4.2 Metric Description

Depending on the problem domain, six region-based segmentation performance metrics are selected for assessing the quality of the segmentation techniques used. It has been mentioned earlier that most of these region-based metrics have been proposed by Kumar and Suhas [11] and Monteiro and Campilho [12], and we have proposed the rest. Some of these metrics are favorable, and the rest of them are adverse. By favorable, we mean the more the metric is, the better result is obtained, and by adverse, we mean the more the metric is, the worse result is obtained. Another property of each of both favorable and adverse metrics is that its value ranges from 0 to 1 inclusive. However, all of the metrics are discussed here.

- Overlap measure (M_O): It is a favorable measure, which is stated as the proportion of the intersection of segmented and ground truth affected areas to the union of segmented and ground truth affected areas. Let segmented affected area be S and ground truth affected area be G. So, the overlap metric is equated as:

Fig. 2 Four frequent diseases of each of three common local fruits in Bangladesh. **a** Guava. **b** Jackfruit. **c** Papaya

$$M_O = \frac{|S \cap G|}{|S \cup G|}. \tag{9}$$

- Under segmentation measure (M_{S-}): It is an adverse measure, which is stated as the proportion of the unsegmented affected area to the ground truth affected area. Let S and G denote segmented affected area and the ground truth affected area, respectively. So, the metric of under segmentation is equated as:

$$M_{S-} = \frac{|G \backslash (S \cap G)|}{|G|} = \frac{|G \backslash S|}{|G|}. \tag{10}$$

- Over segmentation measure (M_{S+}): It is an adverse measure, which is stated as the proportion of segmented unaffected area to the ground truth affected area. Let S and G denote segmented affected area and ground truth affected area, respectively. So, the metric of over segmentation is equated as:

$$M_{S+} = \frac{|S \backslash (G \cap S)|}{|G|} = \frac{|S \backslash G|}{|G|}. \tag{11}$$

- Dice similarity measure (κ): It is a favorable measure, which is conceived from a measure of reliability called kappa statistic [30]. It is stated as the proportion of twice the sum of the intersection of segmented and ground truth affected areas to the sum of segmented and ground truth affected areas. So, the dice similarity measure is equated as:

$$\kappa = \frac{2 \times |G \cap S|}{|G| + |S|}. \tag{12}$$

- Error rate (*ER*): It is an adverse measure, which is stated as the proportion of the sum of the unsegmented affected area and segmented unaffected area to the sum of the segmented affected area and ground truth affected area. Let *S* and *G* denote segmented and ground truth affected areas, respectively. So, the error rate is equated as:

$$ER = \frac{|G \backslash S| + |S \backslash G|}{|G| + |S|}. \tag{13}$$

- Accuracy (A_{cc}): It is a favorable measure, which is stated as the proportion of the sum of segmented and ground truth affected areas except for unsegmented affected area and segmented unaffected area to the sum of segmented and ground truth affected areas. Let *S* and *G* denote segmented affected area and ground truth affected area, respectively. So, the error rate is equated as:

$$A_{cc} = \frac{(|G| + |S|) - (|G \backslash S| + |S \backslash G|)}{|G| + |S|} = 1 - ER. \tag{14}$$

5 Experimental Evaluation

We perform an exhaustive investigation following our approach for analyzing different on-hand segmentation techniques as illustrated in Fig. 1. It commences with the presumption that somebody, i.e., a farmer or a gardener, who is willing to properly recognize local fruit diseases, captures an image of guava, jackfruit, or papaya. Pondering over the distinct people from distinct backgrounds, we have resized the original image inputted. At that point, the original image is changed into a foreordained size image of 300 × 300 pixels. This foreordained size has been set by taking the varying configuration of different cameras into account. Then contrast enhancement is performed by putting color intensity mapping, i.e., histogram equalization into action. Then this image goes through one of the following steps, as shown in Fig. 1, depending on the segmentation algorithm that is going to be used.

(1) A color space conversion from RGB to *L*a*b**
(2) A color space conversion from RGB to gray scale
(3) No conversion.

The image can undergo the color space conversion from RGB to gray scale so that a grayscale histogram can be formed. The image can also go through the color space conversion from RGB to $L*a*b*$, because it has been asserted by Burney and Tariq in [28] that the K-means clustering algorithm outperforms in segmenting images in $L*a*b*$ color space than that in RGB color space. At first, the conversion is performed, as described in [29], from RGB color space to CIE XYZ color space and from XYZ color space to $L*a*b*$ color space. The image is then divided into several regions by applying one of the four prominent image segmentation techniques—region growing, K-means clustering, fuzzy c-means clustering, and Otsu's method. Thus, disease-attacked parts are isolated from disease-free parts. Figure 3 shows the stepwise effects of changes in the images of all disease types. From the attacked portion, a vector containing discriminatory features can be constructed. It has already been mentioned that quality feature extraction depends on the quality segmentation. So, for assessing the quality of the four segmentation techniques, six region-based performance metrics are calculated by putting ground truth images, which are manually labeled, into action. The summary of the stepwise experiments done is provided in Table 1. The results of the performance of the four image segmentation techniques for four diseases of each of the three local fruits along with the ground truth are quantitatively provided in Table 2, but Fig. 4, for the sake of space, pictorially shows the performance results of only the K-means clustering segmentation technique for four diseases of each of the three local fruits along with the ground truth.

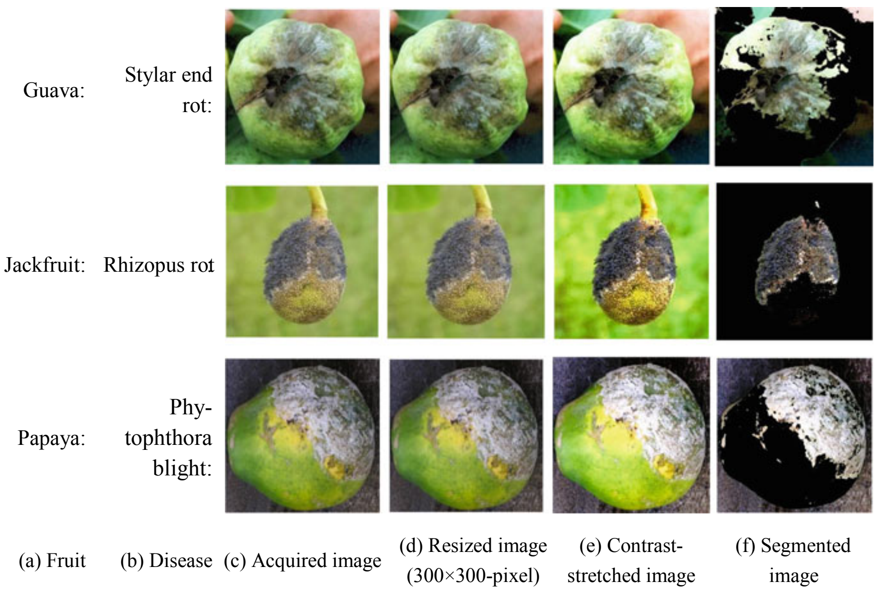


Fig. 3 Stepwise changes of images. **a** Fruit. **b** Disease. **c** Acquired image. **d** Resized image (300 × 300 pixels). **e** Histogram-equalized image. **f** Disease-affected segmented areas

Table 1 Summary of stepwise experiments performed

Experimental stride	Deployed technique	Method/algorithm/parameter used
Preprocessing	Resizing	Bicubic interpolation
	Contrast stretching	Histogram equalization
Segmentation	Region-based	Region growing
	Clustering	*K*-means clustering
		Fuzzy *c*-means clustering
	Thresholding	Otsu's method
Metric calculation	Region-based	Accuracy
		Error rate
		Measure of overlap
		Measure of under segmentation
		Measure of over segmentation
		Dice similarity measure

Table 2 Performance of all segmentation techniques used in the amount of the six performance metrics

Technique	Criterion	Overlap measure (%)	Under segmentation measure (%)	Over segmentation measure (%)	Dice similarity measure (%)	Error rate (%)	Accuracy (%)
Region growing	Guava	66.5	31.44	13.14	79.15	27.09	72.92
	Jackfruit	70.96	13.85	24.03	82.56	20.45	79.55
	Papaya	55.99	15.63	23.52	68.12	26.6	73.4
	Aggregate	64.48	20.31	20.23	76.61	24.71	75.29
K-means clustering	Guava	62.36	17.24	14.91	73.63	22.5	77.5
	Jackfruit	79.60	5.47	12.37	87.81	13.44	86.56
	Papaya	73.35	7.62	17.1	84.85	19.12	80.88
	Aggregate	71.77	10.11	14.79	82.10	18.35	81.65
Fuzzy *c*-means clustering	Guava	63.57	39.31	8.76	76.35	29.78	70.22
	Jackfruit	66.89	26.41	13.23	79.13	25.56	74.44
	Papaya	45.03	3.42	50.72	60.92	37.83	62.17
	Aggregate	58.5	23.05	24.24	72.13	31.06	68.94
Otsu's method	Guava	68.76	18.23	11.86	81.07	22.37	77.63
	Jackfruit	73.9	5.75	14.46	84.73	16.59	83.41
	Papaya	56.44	11.22	41.66	71.56	30.78	69.22
	Aggregate	66.37	11.73	22.66	79.12	23.25	76.75

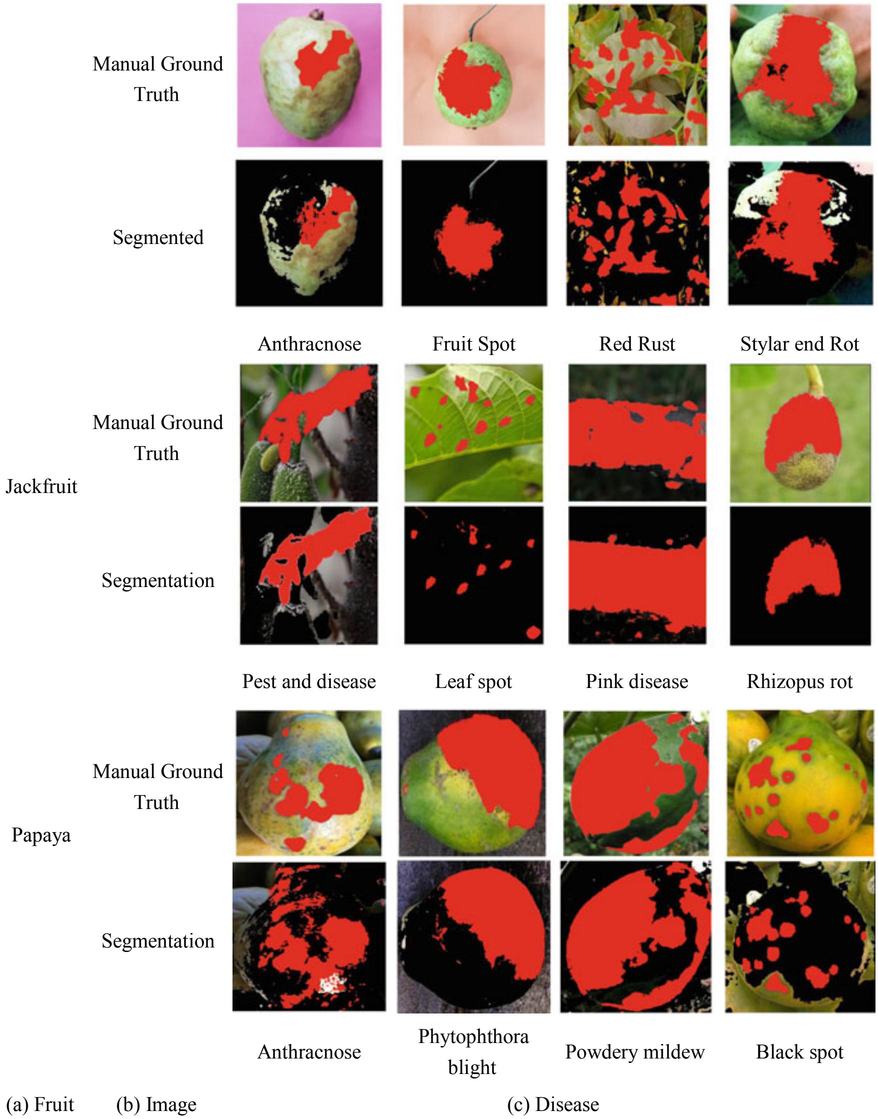


Fig. 4 Performances obtained by K-means clustering segmentation on distinct diseases of three local fruits. **a** Three local fruits. **b** Manual ground truth and segmented images of four diseases of each of three local fruits. **c** Manually outlined affected portions of ground truth images of three local fruits and affected portions of segmented images of three local fruits calculated by applying K-means clustering

We observe from Table 2 that K-means clustering turns out supreme over all other segmentation techniques applied in terms of all performance metrics used other than the two metrics under segmentation measure and over segmentation measure, in cases of which fuzzy c-means clustering outperforms all other segmentation techniques exhibiting the values of 3.42% for papaya and 8.76% for guava, respectively. Thus, K-means clustering not only achieves the maximum number of metric-wise best positions but also shows the best results of metric-wise aggregate values. On the other hand, the aggregate performances of region growing have been the poorest in terms of all metrics used other than the metric over segmentation measure (20.23%). Although the highest accuracy (81.65%) is achieved by K-means clustering, nearby accuracy is exhibited by too. For the six metrics considered, Otsu's method resides just behind the fuzzy c-means clustering leaving the region growing back. We know that accuracy becomes very rigorous if an equal sample size for each class is used, which is very much applicable to our situation. So, we delve deeply into the metric accuracy. Table 3 reveals the accuracy exhibited by each segmentation technique individually for all of the diseases of each of the three local fruits. We notice from Table 3 that the highest accuracy (96.77%) for an individual disease class is exhibited by K-means clustering for the fruit spot of guava. Moreover, it is K-means clustering which exhibits the highest accuracy for each disease of papaya and jackfruit. However, it retains its position, i.e., highest accuracy attainer for all diseases of guava other than red rust, for which region growing acquires the highest accuracy of 87.32%. Taking all of the factors just discussed into account, we can assert that the K-means clustering performs the best of all four segmentation techniques in the context of automated local fruit disease recognition.

6 Conclusion

In this chapter, an exhaustive comparison of four different on-hand image segmentation techniques has been performed on the ground of automated local fruit disease recognition. Sample images of four frequently occurring diseases of each of three common local fruits in Bangladesh, namely papaya, jackfruit, and guava, have been used. All of these sample images are rendered into predetermined size images for comparison on the same ground. K-means clustering has been comprehensively found the best performer attaining an aggregate accuracy of 81.65%. There has been some observation that some inspiring efforts have been shown in automated fruit disease recognition research for the last couple of years. Nevertheless, rigorous segmentation performance assessment focused on meaningful quantitative indications is inadequate yet. Perceiving this situation, our obtained accuracy of 81.65% can be argued to be both good and prospective enough. There exists still enormous future works on Bangladeshi local fruit disease recognition to deal with a much broader range of local fruits and their diseases.

Table 3 Accuracy exhibited by each of the segmentation techniques used

Technique	Accuracy											
	Guava				Jackfruit				Papaya			
	Anthracnose (%)	Fruit spot (%)	Red rust (%)	Stylar end rot (%)	Rhizopus rot (%)	Pest and disease (%)	Leaf spot (%)	Pink disease (%)	Anthracnose (%)	Phytophthora blight (%)	Powdery mildew (%)	Black spot (%)
Region growing	63.21	82.81	87.32	58.32	88.94	71.05	80.83	77.38	79.96	64.48	61.68	87.47
K-means clustering	77.58	96.77	63.06	72.59	93.78	71.22	92.4	88.85	76.27	78.13	84.28	84.85
Fuzzy *c*-means clustering	76.78	88.26	52.2	63.64	94.51	54.27	74.63	74.36	68.04	55.46	38.47	86.71
Otsu's method	66.54	86.67	81.8	75.51	90.44	78.38	81.7	83.11	82.89	62.02	47.27	84.71

References

1. Food security: It's not only about production. https://www.thedailystar.net/supplements/25th-anniversary-special-part-4/food-security-its-not-only-about-production-212035 (2020) Accessed 30 Jan 2020
2. Bangladesh: Employment in agriculture. https://www.theglobaleconomy.com/Bangladesh/Employment_in_agriculture (2020). Accessed 17 May 2020
3. Habib, M. and Rokonuzzaman, M.: Distinguishing feature selection for fabric defect classification using neural network. J. Multimedia. **6**(5) (2011)
4. Gonzalez, R., Woods, R.: Digital Image Processing, 4th edn. Pearson Education Limited, New York, USA (2018)
5. Fruit: Available online. https://en.banglapedia.org/index.php?title=Fruit (2020). Accessed 30 Jan 2020
6. Zhang, H., Fritts, J.E., Goldman, S.A.: Image segmentation evaluation: A survey of unsupervised methods. Comput Vis Image Underst. **110**(2), 260–280 (2008)
7. Shi, R., Ngan, K.N., Li, S., Paramesran, R., Li, H.: Visual quality evaluation of image object segmentation: Subjective assessment and objective measure. IEEE Trans. Image Process. **24**(12), 5033–5045 (2015)
8. Wang, Q., Wang, Z.: September. A subjective method for image segmentation evaluation. In: Asian Conference on Computer Vision, pp. 53–64. Springer, Berlin, Heidelberg (2009)
9. Peng, B., Zhang, L.: October. Evaluation of image segmentation quality by adaptive ground truth composition. In: European Conference on Computer Vision, pp. 287–300. Springer, Berlin, Heidelberg (2012)
10. Unnikrishnan, R., Pantofaru, C., Hebert, M.: September. A measure for objective evaluation of image segmentation algorithms. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)-Workshops pp. 34–34. IEEE (2005)
11. Kumar, Y.S., Suhas, G.: December. Identification and classification of fruit diseases. In: International Conference on Recent Trends in Image Processing and Pattern Recognition pp. 382–390. Springer, Singapore (2016)
12. Monteiro, F.C., Campilho, A.C.: September. Performance evaluation of image segmentation. In: International Conference Image Analysis and Recognition pp. 248–259. Springer, Berlin, Heidelberg (2006)
13. Sahu, S., Sarma, H., Bora, D.J.: August. Image Segmentation and its Different Techniques: An In-Depth Analysis. In 2018 International Conference on Research in Intelligent and Computing in Engineering (RICE) pp. 1–7. IEEE (2018)
14. Lena, F.: https://en.wikipedia.org/wiki/Lena_Fors%C3%A9n (2020). Accessed 19 May 2020
15. Chopaade, P.B., Bhagyashri, K.: February. Image processing based detection and classification of leaf disease on fruits crops. In: 3rd National Conference on Advancements in Communication, Computing and Electronics Technology (ACCET-2016). India, (February, 2016)
16. Rozario, L.J., Rahman, T., Uddin, M.S.: Segmentation of the region of defects in fruits and vegetables. Int. J. Comput. Sci. Inform. Secur. **14**(5), 399 (2016)
17. Wang, L., Li, A., Tian, X.: November. Detection of fruit skin defects using machine vision system. In: 2013 Sixth International Conference on Business Intelligence and Financial Engineering. pp. 44–48. IEEE (2013)
18. Samajpati, B.J., Degadwala, S.D.: Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In: 2016 International Conference on Communication and Signal Processing (ICCSP). pp. 1015–1019. India (2016)
19. Habib, M.T., Majumder, A., Jakaria, A.Z.M., Akter, M., Uddin, M.S., Ahmed, F.: Machine vision based papaya disease recognition. J. King Saud Univ-Comput. Inform. Sci. **32**(3), 300–309 (2020)
20. Habib, M.T., Majumder, A., Nandi, R.N., Ahmed, F., Uddin, M.S.: A comparative study of classifiers in the context of papaya disease recognition. In: Uddin, M., Bansal, J. (eds.) Proceedings of International Joint Conference on Computational Intelligence. Algorithms for Intelligent Systems. Springer, Singapore (2020)

21. Habib, M.T., Mia, M.J., Uddin, M.S., Ahmed, F.: An in-depth exploration of automated jackfruit disease recognition. J. King Saud Univ-Comput. Inform. Sci. (2020)
22. Majumder, A., Habib, M.T., Lima, P.H., Sourav, S., Nandi, R.N.: Automated carrot disease recognition: a computer vision approach. Int. J. Eng. Technol. **7**(4), 5790–5797 (2018)
23. Hosen, M.I., Tabassum, T., Akhter, J., Islam, M.I.: June. Detection of fruits defects using colour segmentation technique. Int. J. Comput. Sci. Inform. Sec. **16**(6), 215–223 (2018)
24. Rachmawati, E., Khodra, M.L., Supriana, I.: Fruit image segmentation by combining color and depth data. AIP Conference Proceedings. AIP Publishing **1746**(1) (2016)
25. Keys, R.: Cubic convolution interpolation for digital image processing. IEEE Trans. Acoust. Speech Signal Process. **29**(6), 1153–1160 (1981)
26. Gonzalez, R., Woods, R.: Digital Image Processing. 3rd ed. Pearson Education Limited (2008)
27. International Telecommunication Union, ITU, Geneva.: "ITU-R Recommendation BT.709-3: Basic Parameter Values for the HDTV Standard for the Studio and for International Programme Exchange" (1998)
28. Burney, S.A., Tariq, H.: K-means cluster analysis for image segmentation. Int. J. Comput. Appl. **96**(4), (2014)
29. Color Conversion Algorithms. https://cs.haifa.ac.il/hagit/courses/ist/Lectures/Demos/ColorApplet2/t_convert.html#RGB%20to%20XYZ%20&%20XYZ%20to%20RGB (2019). Accessed 12 Oct 2019
30. Zou, K.H., Warfield, S.K., Bharatha, A., Tempany, C.M., Kaus, M.R., Haker, S.J., Wells, W.M., III., Jolesz, F.A., Kikinis, R.: Statistical validation of image segmentation quality based on a spatial overlap index1: scientific reports. Acad. Radiol. **11**(2), 178–189 (2004)

Machine Vision-Based Fruit and Vegetable Disease Recognition: A Review

Md. Tarek Habib, Md. Ariful Islam Arif, Sumaita Binte Shorif, Mohammad Shorif Uddin, and Farruk Ahmed

1 Introduction

The world population is growing very rapidly year after year. 100 years ago, the world population was only 1.5 billion, and it stands at about 7.8 billion now [1]. As the number of resources of various productive items are decreasing day by day due to this rapid population growth, this scarcity of resources, especially food, has become a major challenge in meeting the needs of the growing population. According to the food and agriculture organization (FAO), by 2050, the world's population will be about 10 billion, and the consumption of meat, fruits, and vegetables will increase dramatically. Besides, there will be a shortage of natural resources and there will be a lot of pressure on it to meet the demand [2]. It is very important to pay special attention to agriculture to meet the food need of the population. So, agricultural production badly needs to increase. Along with all other enhancements, new technologies need to be used for agro-food systems. Artificial intelligence (AI) can play a very vital role

Md. T. Habib (✉) · S. B. Shorif · M. S. Uddin
Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh
e-mail: md.tarekhabib@yahoo.com

S. B. Shorif
e-mail: sumaita.bs@gmail.com

M. S. Uddin
e-mail: shorifuddin@gmail.com

Md. T. Habib · Md. A. I. Arif
Department of Computer Science and Engineering, Daffodil International University, Dhaka, Bangladesh
e-mail: ar.islam634@gmail.com

F. Ahmed
Department of Computer Science and Engineering, Independent University, Dhaka, Bangladesh
e-mail: farruk60@gmail.com

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_10

in this regard. In particular, it is highly demanded to increase agricultural production by introducing new technologies using AI.

The AI technologies are currently being applied in various fields, including remote sensing scene classification [3], white blood cell classification [4], histopathology image classification [5], deep image esthetics classification [6], food/non-food image classification [7], vehicle classification [8], and textile defect classification [9]. Likewise, AI also has had a special contribution to agriculture, such as fruit and vegetable disease recognition [10–26], fish recognition [27, 28], quality grading of agricultural yields [29, 30], crop disease recognition [31–47], and so on.

In this chapter, we systematically describe state-of-the-art efforts for machine vision-based fruit and vegetable disease recognition. Machine vision-based fruit and vegetable disease recognition involves two main problems, i.e., disease detection and disease classification. At first, we give the readers a clear impression of an expert system for fruit and vegetable disease recognition along with its architecture. Moreover, we exhaustively discuss different computer vision methods used for the recognition of fruit and vegetable diseases to get the readers to have a clear understanding of them. In fine, we thoroughly discuss the state-of-the-art efforts for machine vision-based fruit and vegetable disease recognition and perform a comparative analysis of them comprehensively. This chapter aims to increase the knowledge of those who are interested in working in the field of machine vision-based fruit and vegetable recognition and to help them achieve effective work.

There have a noticeable number of works on machine vision-based fruit and vegetable disease recognition, which is split into two parts, i.e., disease detection and disease classification. Some works have covered the entire recognition process, whereas some jettisoned their works performing detection only. There have been some efforts for reviewing these works. Shruthi et al. [48] have worked on a review of machine learning classification on plant disease detection. In their work, they used five machine learning classifiers, and within them, the convolutional neural network (CNN) provided the most accuracy as well as the highest number of grain disease detection. Alsmadi and Almarashdeh [49] have worked on the fish classification technique. They surveyed preprocessing methods, feature extraction techniques, classification accuracy, and the number of fish species. Dubey and Jalal [50] have worked with image processing applications for fruit and vegetable analysis with different image preprocessing approaches. Bhargava and Bansal [51] have worked for fruits and vegetable equality evaluation with computer vision using different types of algorithms. Naik and Patel [52] have worked on the machine vision-based extraction method, various features, and classifiers for fruit classification and grading. Apart from agriculture, some other reviews were also discussed here.

Ngan et al. [53] have worked with various defect detection approaches and their characteristics. Habib et al. [54] have surveyed and compared the classifiers used on automated fabric deflection classification. Kumar [55] has compared a single approach with a statistical, spectral, model-based approach for computer vision-based fabric defect detection. Egmont-Petersen et al. [56] have worked with the help of a neural network for image preprocessing, data reduction, segmentation, object recognition, image understanding, and optimization. We have divided the work that has been done in agriculture into three parts are disease diagnosis, object recognition, and quality grading. In the case of disease diagnosis, work on fruits was done at [11–15], and work on vegetables was done at [16–18]. Again, in the case of object recognition and detection, work has been done on fruits [19–25], and work on vegetables has been done on [10, 26]. In the quality grading section of organized work in agriculture, work has been done on vegetables [29, 30].

This chapter is organized as follows. Section 1 describes the introduction. Section 2 describes the fruit and vegetable disease recognition basics. Section 3 explains the data set and methods in fruit and vegetable disease recognition. Section 4 contains a performance evaluation metrics. Section 5 explains the state-of-the-art approaches. Section 6 contains the conclusion.

2 Fruit and Vegetable Disease Recognition Basics

Fruit and vegetable recognition is very important in agriculture. Because it is not always possible to test every crop and fruit from the field staff present in the field. Sometimes, the farmer needs immediate control of the disease. Therefore, at this particular time, it is necessary to recognize fruit and vegetable disease using computer vision and machine learning. In this case, the person present in the field can take a picture of the affected fruit or vegetable through his smartphone and send it to the appropriate authority. The image can be sent directly to a manufactured expert system in the form of input and the expert will send a review of the image sent in response to the system: report and results. This process is shown in Fig. 1.

Some processing has to be done to get information directly from the pictures taken on the mobile using artificial intelligence techniques. That image needs to follow certain steps to make an acceptable decision. Image acquisition will be done after taking the image. Then the image will go away for processing. Different methods are followed for processing. After preprocessing, the work will be done in two ways. The first method is to do feature extraction using traditional machine learning. The second step is to classify features by extracting features using neural networks using deep

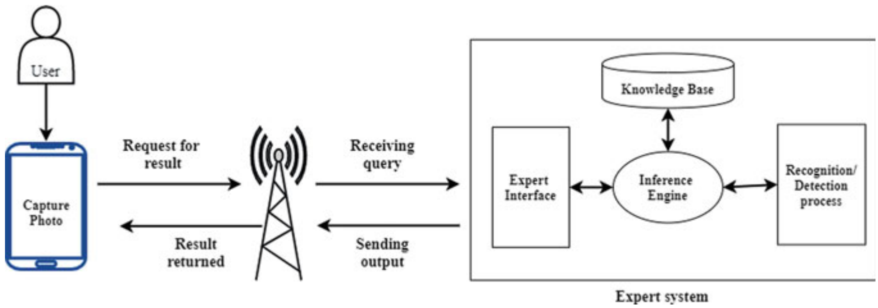


Fig. 1 Architecture of an expert system for vegetable and fruit disease recognition

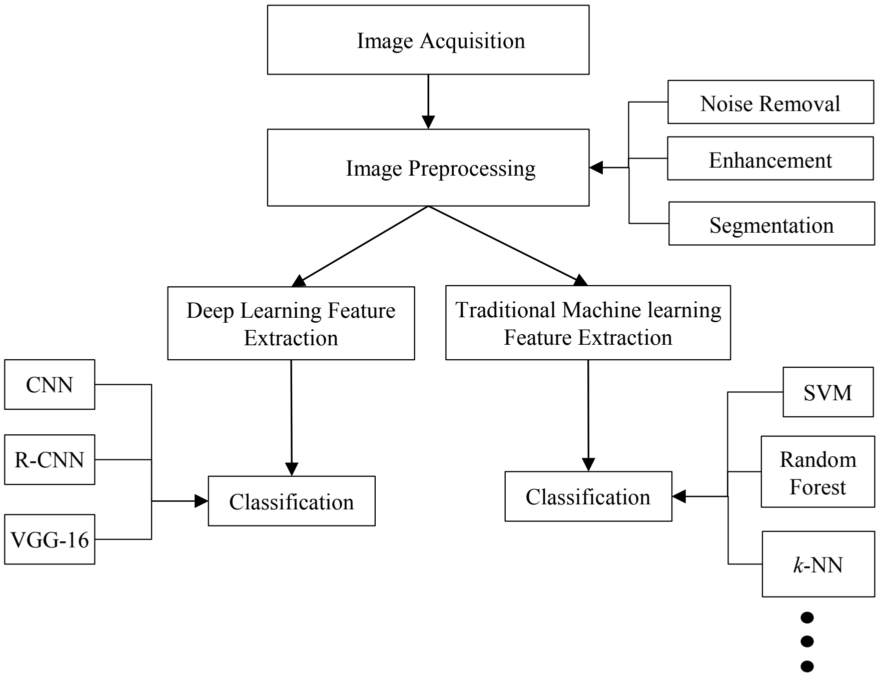


Fig. 2 Steps for the vision-based recognition of fruit and vegetable diseases

learning. According to the classification report received, the decision for disease recognition will be taken and it will be sent to the sender. The whole process is illustrated in Fig. 2.

3 Fruit and Vegetable Disease Recognition Data Set and Methods

Detection and recognition tasks have different difficulties, some of which are determining which algorithm will work properly, which method will be suitable for feature extraction, modeling with app or server, etc. In the case of image classification intra-class variation, scale variation, illumination, viewpoint variation, background cut, etc. are among the difficulties. Object detection from any view involves dealing with multiple views, large pause variations, and resolutions. Color combinations are a problem in image processing at different times, color enhancement is not accurate, problems with photo resizing, not taking pictures from the right angle, etc. The type of image used for image recognition detection as train and test data is shown in Fig. 3. This type of data set can be found on various websites or by taking pictures or by contacting people who have done this before. Many have worked with different data

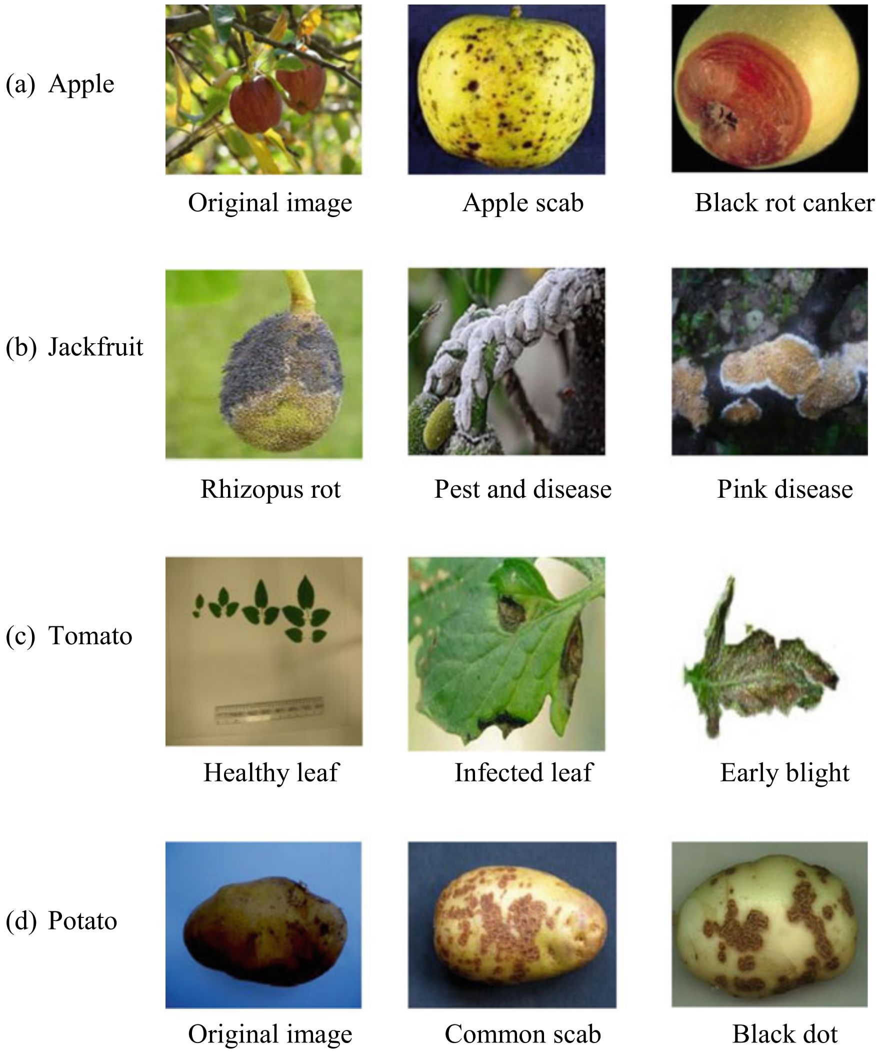


Fig. 3 Different fruit and vegetables diseases. **a** Apple [13, 19]. **b** Jackfruit [14]. **c** Tomato [17]. **d** Potato [13, 29]

sets. Again, many have worked with some common data sets and those data sets are available online [57]. The data sets used by those who have previously worked with fruit and vegetable disease detection and recognition are shown in Table 1.

The function of image or object detection is to process the image using computer technology to bring out the existing object in it. Each image contains more than one different object for object detection. All those objects have to be separated inside the bounding box. In the case of image classification, the localization of each object has to be numerically calculated. Specific pixels are assigned to certain objects. Thus, if

Table 1 Condensation of the data set for machine vision-based fruit and vegetable disease recognition

Name and/or reference	Object containment	Number of classes	Sample size
Fruit 360 [57]	Apple, banana, blueberry, cherry	131	90,483 images
Samajpati and Degadwala [22]	Apple	4	80 images
Omrani et al. [23]	Apple	2	320 samples
Rozario et al. [24]	Apple, banana, tomato, potato	4	63 images
Habib et al. [25]	Jackfruit	5	240 images
Habib et al. [26]	Jackfruit	4	480 images
Vakilian and Massah [27]	Cucumber	2	300 images
Mokhtar et al. [28]	Tomato leaves	2	400 images
Habib et al. [29]	Papaya	5	126 images
Lak et al. [30]	Apple	2	30 images
Zhang and Wu [31]	Apple, banana, grape	18	1653 images
Sa et al. [32]	Apple, avocado, mango	6	54 images
Hossain et al. [33]	Apple, pineapple, papaya, orange	14	5947 images
Zawbaa et al. [34]	Apple, strawberry, orange	3	178 images
Hou et al. [35]	Apple, tomato, orange, kiwi	7	5330 images
Zawbaa et al. [36]	Apple, strawberry, orange	3	178 images
Rocha et al. [37]	Kiwi, plum, cashew	12	32 images
Razmjooy et al. [38]	Potato	2	500 images
Shi and Wu [39]	Tomato	2	*NM*

NM not mentioned

multiple objects are in a picture, those objects, that is, their pixels, are classified into a specific class. All this work is usually done in the image processing section. Different types of machine learning algorithms are used for image preprocessing. These include image size normalization, histogram equalizer, resizing, cropping, median filter and point detection, mutations, etc. Then image segmentation is done using one of the many off-the-shelf techniques. It is needed in traditional machine learning, whereas deep learning, on the other hand, uses deep neural networks directly. Methods used for feature extraction include geometrical, morphological, texture features, moment invariants, distance measurements, landmark points, distance and angle measurements, color histogram, statistical measurement, triangulation technique, optimal separating hyper-plane, the margin of separation, red–green–blue (RGB), and hue–saturation–value (HSV) color features, non-grid part models, wrapper method, etc. [49], whereas some can be easily calculated and some have to be calculated in a slightly more complex way. No such method is required for feature extraction in deep learning as it is processed directly through the neural network.

One of the two methods used in fruit and vegetable disease detection and recognition is the application of traditional machine learning and deep learning. Traditional machine learning can be divided into two parts, one with segmentation and the other without segmentation. This means that at some point after image processing, resizing, and color enhancement, detection is done by looking at the background and the ratio of the object without any further segmentation. In most cases, segmentation is created using background and object positioning, color scale setting, and other ancillary features that help train the machine and provide accurate accuracy. Again, in the case of using deep learning, detection is done by using image processing or directly using the deep neural network without image processing.

Rozario et al. [13] did image segmentation based on gray and color images in their work. They performed this color-based segmentation for detection infected area using *k*-means clustering. Habib et al. [14] transformed the images they collected into a certain size image by using bicubic interpolation and increased the contrast through histogram equalization. Using statistical and gray-level co-occurrence matrix (GLCM) features, they separated the affected area and the healthy area. On the other hand, Hossain et al. [22] did the work of fruit classification using deep learning. They cropped the images during image processing, keeping height and width equal. Feature transformation was then done through repeated convolution and pooling operations.

4 Performance Evaluation Metrics

We have already mentioned that machine vision-based tasks are performed in two ways. One through detection, the other through recognition. Since two approaches are followed for recognition and detection, their calculations are divided into two. One is the calculation for feature extraction or segmentation and the other is classification performance calculation.

Localization error was used to measure segmentation performance in the works done in [14, 18]. Localization error is an adverse measure, which is stated as the proportion of the sum of the unsegmented affected area and segmented unaffected area to the sum of the segmented affected area and ground truth affected area. If the segmented area is expressed with A and the ground truth affected area is expressed with B, the localization error (E) can be expressed in the following way:

$$E = \frac{A \sim B}{A + B} \times 100\% \tag{1}$$

Here can be mentioned to the readers that normalized probabilistic rand (NPR), bidirectional consistency error, Hamming distance—these metrics can also be used to measure segmentation performance [58–60]. Nonetheless, their uses have not been observed in any work. Measuring the performance of disease segmentation has been disregarded in almost all works done so far.

Accuracy, sensitivity, specificity, precision, recall, and F_1-score are some conspicuous metrics for measuring the performance of classification [61]. If TP, TN, FP, and FN indicate the number of true positives, true negatives, false positives, and false negatives, respectively, in a confusion matrix for a binary, i.e., two-class problem, these performance metrics can be equated as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100\% \tag{2}$$

$$\text{Sensitivity} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \tag{3}$$

$$\text{Specificity} = \frac{\text{TN}}{\text{FP} + \text{TN}} \times 100\% \tag{4}$$

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \times 100\% \tag{5}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \times 100\% \tag{6}$$

$$F_1\text{-score} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \times 100\% \tag{7}$$

Here is to mention the readers that the detailed calculation of these metrics for a multiclass, i.e., more than two-class problems, where the resulting confusion matrix if of dimension $n \times n$ ($n > 2$) can be found in [18]. However, these conspicuous metrics, especially accuracy, have been widely used in fruit and vegetable disease classification.

5 State-of-the-Art Approaches

The use of computer vision and machine learning for vegetable and fruit disease detection and recognition is increasing day by day. In this case, it is seen that many people use the same method for feature extraction, segmentation, and classifier, while many others follow different apps. Here, we discuss the application of different methods and their results. Apple's detection and classification work has been done using random forest classifiers. In that work, image segmentation has been done using the k-means clustering algorithm and feature extraction has been done according to apple's color, texture, and shape [11]. That work has been used to diagnose apple disease with the help of leaves. The leaves are photographed under specific light control and the infected areas are identified using *k*-means clustering. Three classifications were then applied in which support vector machine (SVM) gave the best performance [12].

Work has been done with a computer vision approach for the extraction of affected areas of fruits and vegetables. Here, the affected areas have been extracted using the Otsu method, background removal of the image has been done, and segmentation has been done using *k*-means clustering [13]. In that research, models are made using computer vision to diagnose jackfruit disease. Here, all the images are converted to a certain size and its contrast is fixed through histogram equalization as well as defective separation using GLCM. Three classifiers have been used to diagnose jackfruit disease of which SVM has achieved 88.67% accuracy [14]. Automated models for the diagnosis of jackfruit are developed, where images are first resized, contrast settings are set, and colors are converted from RGB to $L^*a^*b^*$. Here, five jackfruit diseases have been worked on and nine classifiers have been used of which random forest has been able to achieve 89.59% [15].

Cucumber fungal disease detection is done by looking at the symptoms of cucumber leaves. In this case, the backpropagation artificial neural network (ANN) is used as a classifier. Images collected with three textual features and two thermal parameters are normalized and measured [16]. An approach is used to detect and identify unhealthy tomato leaves using image processing techniques, where 99.83% accuracy is achieved with the help of the SVM algorithm using a linear kernel function. The GLCM method was used for image processing and feature extraction, and the n-fold cross-validation technique was used for classifier performance [17]. Work was done for papaya disease recognition with the help of a machine vision-based agro-medical expert system. After image segmentation using *k*-means clustering, recognition was done based on six different diseases. After using three machine learning classifiers, it was found that SVM provides 90.15% accuracy [18].

The recognition of apple fruit was done with the help of natural luminance using the machine vision, where segmentation was done by edge detection and color and shape analysis. Color–shape-based algorithm got 83.33% accuracy in apple detection [19]. Fruit classification was done with the help of computer vision using a multiclass kernel support vector machine, where backgrounds are removed using split and merge algorithm, feature spaces are composed using color histograms, texture features, and dimensions of feature spaces are reduced using principal component analysis (PCA) [20]. Fruit detection was done using a deep convolutional neural network, where faster region-based CNN is used the color (RGB) and near-infrared (NIR) modalities [21].

Automatic classification is used using deep learning. They work with light architecture and visual geometry group (VGG-16) based architecture model [22]. Automatic fruit classification was done using random forest algorithms and processed by scale-invariant feature transform (SIFT) using shape, color characteristics [23]. Here, the images were given directly to the neural network as input so the processes of feature extraction or data reconstruction did not need to be done here. Everything was done inside the convolution layer, the pooling layer, and the full-connected layer [24]. Automatic fruit image recognition is based on shape and color features. During preprocessing, 90×90 pixels images are used for creating a data set and applied to the *k*-nearest neighbor (*k*-NN) and SVM algorithm classification [25].

Their research worked with CNN using fruit and vegetable classification systems. Images of fruit and vegetable classification were trained using the VGG model. Besides, feature extraction and classification of images using CNN models were implemented [10]. They worked with automatic fruit and vegetable classifications from images, where *k*-means classifier used and global color histogram used from preprocessing [26]. In that research, the inspection method was worked with the help of machine learning. The contrast was increased after the images were taken at a certain resolution. The RGB intensity color model was then used for area detection and thresholding [29]. That research studied tomato processing defect detection using deep learning. Images were analyzed using the deep learning network and mask R-CNN machine learning framework feature extraction of tomato images [30].

All the data has been collected in one table for comparative analysis of the work done so far by applying computer vision and mission learning with any kind of detection and recognition related to agro-food-related. It is shown in Table 2.

6 Conclusion

In this chapter, we have analyzed and discussed thoroughly the different methods of vision-based fruit and vegetable disease recognition. The entire task of recognition is branched into two divisions—one is with the deployment of traditional machine learning and another is with deep learning. We have found image resizing, gray-scale imaging, GLCM calculation, etc. some important steps for vision-based fruit and vegetable disease recognition. We have also found *k*-means clustering segmentation a very popular and common method for fruit and vegetable disease segmentation. Diseases have been classified by different prominent classifiers, where ANN, *k*-NN, SVM are the most popular and common ones. CNNs have also been used for fruit and vegetable recognition. The exhaustive discussion of this chapter will render a proper understanding and a good insight to those researchers who are interested in working and conducting research on machine vision-based fruit and vegetable disease recognition shortly.

Table 2 Comparison of related work of vision-based fruit and vegetable recognition

Method/work done	Object(s) dealt with	Solution domain	Segmentation algorithm	Classifier	Sample size	Accuracy
Samajpati and Degadwala [22]	Apple	Traditional machine learning	*k*-means clustering	Random forest	80 images	*NM*
Omrani et al. [23]	Apple	Traditional machine learning	*k*-means clustering	SVM	320 samples	*NM*
Habib et al. [25]	Jackfruit	Traditional machine learning	*k*-means clustering	SVMs	240 images	88.67%
Habib et al. [26]	Jackfruit	Traditional machine learning	*k*-means clustering	Random forest	480 images	89.59%
Vakilian and Massah [27]	Cucumber	Deep learning	*NA*	Backpropagation ANN	300 images	*NM*
Mokhtar et al. [28]	Tomato leaves	Traditional machine learning	Gradient clustering algorithm	SVM	400 images	99.83%
Habib et al. [29]	Papaya	Traditional machine learning	*k*-means clustering	SVM	126 images	90.15%
Lak et al. [30]	Apple	Traditional machine learning	Color–shape-based algorithm	Color–shape-based algorithm	0 images	83.33%
Zhang and Wu [31]	Fruit	Traditional machine learning	Split and merge algorithm	Max-wins-voting SVM	1653 images	88.20%
Sa et al. [32]	Fruit	Deep learning	*NA*	R-CNN	54 images	83%
Hossain et al. [33]	Fruit	Deep learning	*NA*	VGG-16	32 images	*NM*

(continued)

Table 2 (continued)

Method/work done	Object(s) dealt with	Solution domain	Segmentation algorithm	Classifier	Sample size	Accuracy
Zawbaa et al. [34]	Fruit	Traditional machine learning	SIFT	Random forest	178 images	96.97%
Hou et al. [35]	Fruit	Deep learning	*NA*	CNN	5330 images	99.77%
Zawbaa et al. [36]	Fruit	Traditional machine learning	*k*-NN	SVM	178 images	90.91%
Razmjooy et al. [38]	Potato	Traditional machine learning	*NM*	SVM	500 images	95%
Shi and Wu [39]	Tomato	Deep learning	ResNet-101	Mask R-CNN	*NM*	*NM*

NM not mentioned
NA not applicable

References

1. Current World Population: Available at https://www.worldometers.info/world-population/#ref-1. Accessed 25 Aug 2020
2. FAO: The Future of Food and Agriculture—Trends and Challenges. Rome (2017). ISBN 978-92-5-109551-5
3. Zhang, J., Lu, C., Li, X., Kim, H., Wang, J.: A full convolutional network based on DenseNet for remote sensing scene classification. Math. Biosci. Eng. **16**(5), 3345–3367 (2019)
4. Habibzadeh, M., Jannesari, M., Rezaei, Z., Baharvand, H., Totonchi, M.: Automatic white blood cell classification using pre-trained deep learning models: ResNet and Inception. In: Proceedings SPIE 10696, Tenth International Conference on Machine Vision (ICMV 2017), 1069612, 13 Apr 2018
5. Kieffer, B., Babaie, M., Kalra, S., Tizhoosh, H.R.: Convolutional neural networks for histopathology image classification: training vs. using pre-trained networks. In: 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, pp. 1–6 (2017)
6. Jin, X., Chi, J., Peng, S., Tian, Y., Ye, C., Li, X.: Deep image aesthetics classification using inception modules and fine-tuning connected layer. In: 2016 8th International Conference on Wireless Communications & Signal Processing (WCSP), Yangzhou, pp. 1–6 (2016)
7. Singla, A., Yuan, L., Ebrahimi, T.: Food/non-food image classification and food categorization using pre-trained GoogLeNet model. In: Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management (MADiMa'16). Association for Computing Machinery, New York, NY, USA, pp. 3–11 (2016)
8. Jung, H., Choi, M., Jung, J., Lee, J., Kwon, S., Jung, W.Y.: ResNet-based vehicle classification and localization in traffic surveillance systems. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, pp. 934–940 (2017)
9. Habib, M.T.: Machine Vision Based Textile Defects Classification. LAP Lambert Academic Publishing (2017)
10. Zeng, G.: Fruit and vegetables classification system using image saliency and convolutional neural network. In: 2017 IEEE 3rd Information Technology and Mechatronics Engineering Conference (ITOEC), Chongqing, pp. 613–617 (2017)
11. Samajpati, B.J., Degadwala, S.D.: Hybrid approach for apple fruit diseases detection and classification using random forest classifier. In: 2016 International Conference on Communication and Signal Processing (ICCSP), Melmaruvathur, pp. 1015–1019 (2016)
12. Omrani, E., Khoshnevisan, B., Shamshirband, S., Saboohi, H., Anuar, N.B., Md Nasir, M.H.N.: Potential of radial basis function-based support vector regression for apple disease detection. Measurement **55**, 512–519 (2014)
13. Rozario, L.J., Rahman, T., Uddin, M.S.: Segmentation of the region of defects in fruits and vegetables. Int. J. Comput. Sci. Inf. Secur. **14**(5), 399–406 (2016)
14. Habib, M., Mia, M., Mia, M., Uddin, M., Ahmed, F.: A computer vision approach for jackfruit disease recognition. In: Uddin, M., Bansal, J. (eds.) Proceedings of International Joint Conference on Computational Intelligence. IJCCI 2019. Algorithms for Intelligent Systems. Springer, Singapore (2020)
15. Habib, M.T., Mia, M.J., Uddin, M.S., et al.: An in-depth exploration of automated jackfruit disease recognition. J. King Saud Univ. Comput. Inf. Sci. (2020)
16. Vakilian, K.A., Massah, J.: An artificial neural network approach to identify fungal diseases of cucumber (*Cucumis sativus* L.) plants using digital image processing. Arch. Phytopathol. Plant Prot. **46**(13), 1580–1588 (2013)
17. Mokhtar, U., El-Bendary, N., Hassenian, A.E., Emary, E., Mahmoud, M.A., Hefny, H., Tolba, M.F.: SVM-based detection of tomato leaves diseases. In: Filev, D., et al. (eds.) Intelligent Systems'2014. Advances in Intelligent Systems and Computing, vol. 323. Springer, Cham (2015)

18. Habib, M.T., Majumder, A., Jakaria, A.Z.M., Akter, M., Uddin, M.S., Ahmed, F.: Machine vision based papaya disease recognition. J. King Saud Univ. Comput. Inf. Sci. **32**(3), 300–309 (2020)
19. Lak, M.B., Minaei, S., Amiriparian, J., Beheshti, B.: Apple fruits recognition under natural luminance using machine vision. Adv. J. Food Sci. Technol. **2**(6), 325–327 (2010)
20. Zhang, Y., Wu, L.: Classification of fruits using computer vision and a multiclass support vector machine. Sensors (Basel) **12**(9), 12489–12505 (2012)
21. Sa, I., Ge, Z., Dayoub, F., Upcroft, B., Perez, T., McCool, C.: DeepFruits: a fruit detection system using deep neural networks. Sensors **16**(8), 1222 (2016)
22. Hossain, M.S., Al-Hammadi, M., Muhammad, G.: Automatic fruit classification using deep learning for industrial applications. IEEE Trans. Ind. Inform. **15**(2), 1027–1034 (2019)
23. Zawbaa, H.M., Hazman, M., Abbass, M., Hassanien, A.E.: Automatic fruit classification using random forest algorithm. In: 2014 14th International Conference on Hybrid Intelligent Systems, Kuwait, pp. 164–168 (2014)
24. Hou, L., Wu, Q., Sun, Q., Yang, H., Li, P.: Fruit recognition based on convolution neural network. In: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), Changsha, pp. 18–22 (2016)
25. Zawbaa, H.M., Abbass, M., Hazman, M., Hassenian, A.E.: Automatic fruit image recognition system based on shape and color features. In: International Conference on Advanced Machine Learning Technologies and Applications, Nov 2014, pp. 278–290. Springer, Cham (2014)
26. Rocha, A., Hauagge, D.C., Wainer, J., Goldenstein, S.: Automatic fruit and vegetable classification from images. Comput. Electron. Agric. **70**(1), 96–104 (2010)
27. Sharmin, I., Islam, N.F., Jahan, I., Joye, T.A., Rahman, M.R., Habib, M.T.: Machine vision based local fish recognition. SN Appl. Sci. **1**, 1529 (2019)
28. Junayed, M.S.: Deep learning based approach for identification of local fish. Doctoral dissertation, Daffodil International University (2018)
29. Razmjooy, N., Mousavi, B.S., Soleymani, F.: A real-time mathematical computer method for potato inspection using machine vision. Comput. Math. Appl. **63**(1), 268–279 (2012)
30. Shi, X., Wu, X.: Tomato processing defect detection using deep learning. In: 2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, pp. 728–732 (2019)
31. Hossain, S., Mou, R.M., Hasan, M.M., Chakraborty, S., Razzak, M.A.: Recognition and detection of tea leaf's diseases using support vector machine. In: 2018 IEEE 14th International Colloquium on Signal Processing & its Applications (CSPA), Batu Feringghi, pp. 150–154 (2018)
32. Naikwadi, S., Amoda, N.: Advances in image processing for detection of plant diseases. Int. J. Appl. Innov. Eng. Manag. (IJAIEM) **2**(11) (2013)
33. Ramesh, L., Nirmala, V.: Betel leaf disease classification using clustering method. J. Chem. Pharm. Sci. Spec. Issue **1**, 193–196 (2017)
34. Jayanthi, S.K., Lalitha, C.: Betel leaf disease detection using histogram of oriented gradients and multiclass SVM. Int. J. Innov. Res. Comput. Commun. Eng. **5**(8), 13994–14001 (2017)
35. Tamilsankar, P., Gunasekar, T.: Computer aided diseases identification for betel leaf. Int. Res. J. Eng. Technol. **2**(9), 2577–2581 (2015)
36. Parikh, A., Raval, M.S., Parmar, C., Chaudhary, S.: Disease detection and severity estimation in cotton plant from unconstrained images. In: 2016 IEEE International Conference on Data Science and Advanced Analytics (DSAA), Montreal, QC, pp. 594–601 (2016)
37. Vijayakumar, J., Arumugam, D.S.: Early detection of powdery mildew disease for betelvine plants using digital image analysis. Int. J. Mod. Eng. Res. **2**(4), 2581–2583 (2012)
38. Dey, A.K., Sharma, M., Meshram, M.R.: Image processing based leaf rot disease, detection of betel vine (Piper Betle L.). Procedia Comput. Sci. **85**, 748–754 (2016)
39. Kurniawati, N.N., Abdullah, S.N.H.S., Abdullah, S., Abdullah, S.: Investigation on image processing techniques for diagnosing paddy diseases. In: 2009 International Conference of Soft Computing and Pattern Recognition, Malacca, pp. 272–277 (2009)

40. Majid, K., Herdiyeni, Y., Rauf, A.: I-PEDIA: mobile application for paddy disease identification using fuzzy entropy and probabilistic neural network. In: 2013 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Bali, pp. 403–406 (2013)
41. Batule, V.B., Chavan, G.U., Sanap, V.P., Wadkar, K.D.: Leaf disease detection using image processing and support vector machine (SVM). J. Res. **2**(02) (2016)
42. Naik, M.R., Sivappagari, C.M.R.: Plant leaf and disease detection by using HSV features and SVM classifier. Int. J. Eng. Sci. Comput. **6**(12), 3794–3797 (2016)
43. Feng, Y., Ren, G., He, K., Liu, Y., Li, L.: RGB color channel variation based segmentation of crop leaf lesion. In: 2015 IEEE 10th Conference on Industrial Electronics and Applications (ICIEA), Auckland, pp. 592–596 (2015)
44. Latte, M.V., Shidnal, S., Anami, B.S.: Rule based approach to determine nutrient deficiency in paddy leaf images. Int. J. Agric. Technol. **13**(2), 227–245 (2017)
45. Abdullah, N.E., Rahim, A.A., Hashim, H., Kamal, M.M.: Classification of rubber tree leaf diseases using multilayer perceptron neural network. In: 2007 5th Student Conference on Research and Development, Selangor, Malaysia, pp. 1–6 (2007)
46. Barbedo, J.G.A.: Digital image processing techniques for detecting, quantifying and classifying plant diseases. SpringerPlus **2**(1), 660 (2013)
47. Vijayakumar, J., Arumugam, D.S.: Recognition of powdery mildew disease for betelvine plants using digital image processing. Int. J. Distrib. Parallel Syst. **3**(2), 231 (2012)
48. Shruthi, U., Nagaveni, V., Raghavendra, B.K.: A review on machine learning classification techniques for plant disease detection. In: 2019 5th International Conference on Advanced Computing & Communication Systems (ICACCS), Coimbatore, India, pp. 281–284 (2019)
49. Alsmadi, M.K., Almarashdeh, I.: A survey on fish classification techniques. J. King Saud Univ. Comput. Inf. Sci. (2020)
50. Dubey, S., Jalal, A.: Application of image processing in fruit and vegetable analysis: a review. J. Intell. Syst. **24**(4), 405–424 (2015)
51. Bhargava, A., Bansal, A.: Fruits and vegetables quality evaluation using computer vision: a review. J. King Saud Univ. Comput. Inf. Sci. (2018)
52. Naik, S., Patel, B.: Machine vision based fruit classification and grading—a review. Int. J. Comput. Appl. **170**(9), 22–34 (2017)
53. Ngan, H.Y.T., Pang, G.K.H., Yung, N.H.C.: Automated fabric defect detection—a review. Image Vis. Comput. **29**(7), 442–458 (2011)
54. Habib, M.T., Faisal, R.H., Rokonuzzaman, M., Ahmed, F.: Automated fabric defect inspection: a survey of classifiers. ArXiv, abs/1405.6177 (2014)
55. Kumar, A.: Computer-vision-based fabric defect detection: a survey. IEEE Trans. Ind. Electron. **55**(1), 348–363 (2008)
56. Egmont-Petersen, M., de Ridder, D., Handels, H.: Image processing with neural networks—a review. Pattern Recogn. **35**(10), 2279–2301 (2002)
57. Fruit 360: Available at https://www.kaggle.com/moltean/fruits. Accessed 07 Aug 2020
58. Unnikrishnan, R., Pantofaru, C., Hebert, M.: A measure for objective evaluation of image segmentation algorithms. In: Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), USA, Sept 2005
59. Shi, R., Ngan, K.N., Li, S., Paramesran, R., Li, H.: Visual quality evaluation of image object segmentation: subjective assessment and objective measure. IEEE Trans. Image Process. **24**(12), 5033–5045 (2015)
60. Monteiro, F.C., Campilho, A.C.: Performance evaluation of image segmentation. In: Image Analysis and Recognition. ICIAR 2006. Lecture Notes in Computer Science, vol. 4141, pp. 248–259. Springer, Berlin, Heidelberg (2006)
61. Han, J., Kamber, M., Pei, J.: Data Mining Concept and Technique, 3rd edn., pp. 332–398. Morgan Kaufmann (2012)

An Efficient Bag-of-Features for Diseased Plant Identification

Raju Pal, Himanshu Mittal, Avinash Pandey, and Mukesh Saraswat

1 Introduction

Being the part and parcel of human beings, agriculture is the main contributing factor in the economy of many countries [1]. The variety of plants harvested in a region depend upon the environmental and land conditions. Additionally, the quantity and quality of the harvested plant is dependent on a number of factors like water supply, natural calamities, plant diseases, etc. [2]. To mitigate the same, a farmer may be aided with advanced vision technologies to take preventive measures [3]. One such technology is automated diseased plant identification system through leaf images [4]. Such a identification system will advantage farmers in number of ways like timely update about plant disease, increase in plant productivity, and minimum resource wastage without the involvement of an expert [5]. In literature, computer vision techniques have been able to detect plant ailments such as yellow or brown spots, primary or late blister, caused by bacteria, virus, or fungus [6]. To do so, leaves are the primary source of disease identification in plants [7]. However, each leaf of a plant presents large disparity in size, texture, color, and shape, which make the disease classification a complex and challenging task. Therefore, this chapter presents an efficient method for the diseased plant identification using leaf images.

R. Pal · H. Mittal · A. Pandey · M. Saraswat (✉)
Jaypee Institute of Information Technology, Noida, India
e-mail: saraswatmukesh@gmail.com

R. Pal
e-mail: raju3131.pal@gmail.com

H. Mittal
e-mail: himanshu.mittal224@gmail.com

A. Pandey
e-mail: avish.nsit@gmail.com

M. S. Uddin and J. C. Bansal (eds.), *Computer Vision and Machine Learning in Agriculture*, Algorithms for Intelligent Systems,
https://doi.org/10.1007/978-981-33-6424-0_11

Generally, the accuracy of an image identification method is largely related to the features extracted by the system [8]. In the literature, there are number of feature extraction methods which can be categorized into two classes [9], namely classical and deep features. In classical features, different attributes of an image are measured, such as shape, size, or distribution of pixels to extract relatively complex descriptors which are rich representation of interest points in the image. Some of the popular methods in this category are scale-invariant features transform (SIFT) [10], speeded-up robust features (SURF) [11], and oriented FAST and rotated BRIEF (ORB) [12]. The second category, deep-level features, is the image descriptors extracted through different machine learning models like convolutional neural network (CNN) [13], inception [14], and others. However, these feature extraction methods require a large dataset along with high computational resources. In literature, classical feature descriptors have produced promising results, especially when used with bag-of-features (BOF) [15, 16]. BOF [15] is a popular image classification method which is based on the concept of bag-of-words framework of text classification. Csurka et al. [17] applied this method for image categorization while Mittal and Saraswat [18] employed the BOF for the classification of tissue images.

Generally, the BOF method includes mainly four phases, namely feature extractor, codebook construction, feature encoding, and classification [19]. In the first phase, feature extraction method is used to extract the keypoints from the training set. Some of the commonly used feature extraction methods are SIFT, SURF, and ORB. Table 1 [20] briefs the existing feature extraction methods of BOF. In the table, methods are categorized into interest-point-based features and learned features. The interest-point detection-based methods detect corner points in an image to extract the features while learned features correspond to deep features which represent high-level abstraction of an image. In literature, it has been witnessed that accuracy of a BOF method is largely dependent on the extracted descriptors. However, BOF method considers only single-feature extraction method which limits the variability in considered features. To alleviate the same, this chapter presents a new BOF method which considers different feature descriptors at a time to generate an efficient visual word vocabulary.

Table 1 Categorization of features extraction methods used in bag-of-features [20]

Category	Classification	Comments	Methods
Blob detection (interest point)	Partial differential equations (PDE) based	Use the PDE on Guassian-scale spaces	DoG [21], Hessian–Laplacian [22], SIFT [23], SURF [11]
	Template based	Decision tree and binary comparison methods are used; computationally fast	BRISK [24], FREAK [25], ORB [12]
Learned features	Deep features	High-level abstractions obtained from raw images	SSAE [26], CNN [27]

In codebook construction, clustering methods [28] like k-means and fuzzy cmeans are employed which quantize extracted features as vocabulary of visual words. In literature, various clustering methods have been used for codebook construction and Table 2 groups these methods into three classes, namely hierarchical methods, partitional methods, and meta-heuristic-based methods. The hierarchical and partitional methods have reported number of demerits like biased clustering results, dependence over initial parameters setting, and varying results with varying number of clusters. Therefore, researchers have introduced meta-heuristic algorithms for efficient clustering [29–31]. Meta-heuristic algorithms are the mathematical models which mimic the optimization behavior of nature [32–35]. These algorithms have been efficient in solving many real-world problems belonging to different domains such as medical image analysis, segmentation [36, 37], software testing, wireless sensor networks [38, 39], data classification [40], and many others. However, all such algorithms have non-deterministic behavior [41] and computationally expensive. Therefore, this paper performs clustering through a popular deterministic algorithm, i.e., gray relational analysis [42].

Further, the feature encoding phase utilizes the generated vocabulary of visual words to encode an image as a vector. Each value of vector represents the count of visual words. The different encoding methods can be divided into three categories, namely voting, reconstruction, and super-vector-based encoding methods [44] which are enlists in Table 3.

Lastly, the encoded features of training images are used to train a classifier which infers label for new images. The performance of a classifier is highly affected by the quality of encoded features.

The overall contribution of the paper can be briefed in following three folds.

1. Weighted two-dimensional quantization method is used for image encoding which uses dual features for encoding the image.

Table 2 Various codebook construction methods used in the bag-of-features [43]

Category	Methods	Comments
Hierarchical methods	Agglomerative clustering, mean shift	These methods can not be applied to large datasets or histology images due to high computational cost
Partitional methods	K-means, FCM, GMM	Generates non-uniform coding (biased toward dense regions); non robust; optimal codebook size (K) is unambiguous
Meta-heuristic-based methods	PSO, GSA, WOA	Used to find optimal visual words based on some objective function defined over compactness and separation; computationally expensive

Table 3 Various types of feature encoding methods [44, 45]

Type	Methods	Comments
Voting based	HV [15], SA [46], LSA [47], salient coding [48],	Based on the formation of histograms which represent the distribution of visual words
Reconstruction based	OMP [49], Sparse coding [50], LLC [51], LCC [52]	Each feature should be reconstructed by the visual words by applyin some constraints and solving least-sqaure optimization problem
Super vector based	LTC [53], SVC [54], VLAD [55]	Gaussian mixture model is used for estimating the feature distributions which contain means, co-variance, and weights of Guassian ditributions

2. For efficient codebook construction, gray relational analysis-based clustering method is used.
3. The proposed BoVW method is leveraged on the diseased plant identification using leaf images, taken from a publicly available dataset, namely PlantVillage dataset [56].

In this chapter, the proposed diseased plant identification uses three different feature extraction methods, namely SIFT, SURF, and ORB which are discussed in the following sections.

1. **Scale-Invariant Feature Transform**: SIFT detects and describes the low-level features from the digital images. It finds some interest points and then represents them in a quantitative manner known as descriptors. These descriptors are invariant to scale, rotation, and illumination conditions. The process of feature extraction and representation is performed in four steps. First, the locations of interest points or keypoints are approximated. These keypoints are the extreme values of scale space pyramid generated through difference of Gaussian images. Second, the keypoint locations are refined. Third, the orientation assignment is performed based on the orientation histogram. And finally, each keypoint is represented by a 128 dimensional vector. Raza et al. [57] studied and analyzed the rotation and scale invariance behavior in the bag-of-features method for the classification of histopathological images.
2. **Speeded-up Robust Features**: SURF [11] is an efficient feature detector method that finds integrals of an image to perform convolution. It includes three phases, namely keypoint detector, feature description, and feature matching. To detect a keypoint, the determinant of the Hessian matrix is calculated around the considered keypoint. If the determinant is maximal, the corresponding keypoint is

selected. In the feature description phase, the detected keypoints are depicted as the intensity distribution of the neighborhood pixels. Lastly, feature matching is performed between different pairs of feature descriptors. Wand and Chen [58] addressed the problem of image alignment in medical or clinical diagnosis. The authors have considered two types of medical images, namely tissue and X-ray images and compared with five other methods, namely TrackEM2, UnwarpJ, BUnwarpJ, mutual information, and SURF. The proposed method shows better results on both of the datasets.

3. **Oriented FAST and Rotated BRIEF**: ORB [12] leverages the strengths of two efficient feature extraction methods, namely FAST [59] and BRIEF [60]. In this, oriented FAST method detects all the keypoints by employing a circular ring. However, it fails to detect the corner points. Moreover, ORB considers only top N keypoints by sorting the FAST keypoints on the basis of Harris corner parameter. Further, multi-scale FAST features are computed at different levels of pyramid scale. To calculate corner orientation, centroid intensity approach is followed while color is ignored. However, the description of a keypoint is generated through rotated BRIEF [60]. This is a rotational invariant version of the BRIEF method which incorporates a learning step. The feature vector is described in terms of patch and rotation matrix. The above-mentioned feature extraction methods are being used to model the new diseased plant identification system which is presented in the following section.

2 Proposed Method

In this work, an automated diseased plant identification system based on the enhanced BOF method is proposed. The block diagram of the developed system is shown in Fig. 1. This system consists of four phases, (i) feature extraction from histopathological images, (ii) the features are clustered using gray relational analysis-based clustering method to generate codebooks, (iii) each image is encoded using the two dimensional vector quantization method, and (iv) the encoded images along with the labels are used to train the classifier which is further used to predict the labels of validated images. The detail description of each phase of the system is presented in the following sections.

2.1 Feature Extraction

The first phase of the new system is the feature extraction. In standard BOF method, only one feature extraction method is used while in the proposed system, combination of two feature extraction methods is used. This results in achieving rich and detailed feature description of the plant images. In this work, combination of any two feature extraction methods is used. For the same, different combinations of SIFT, SURF,

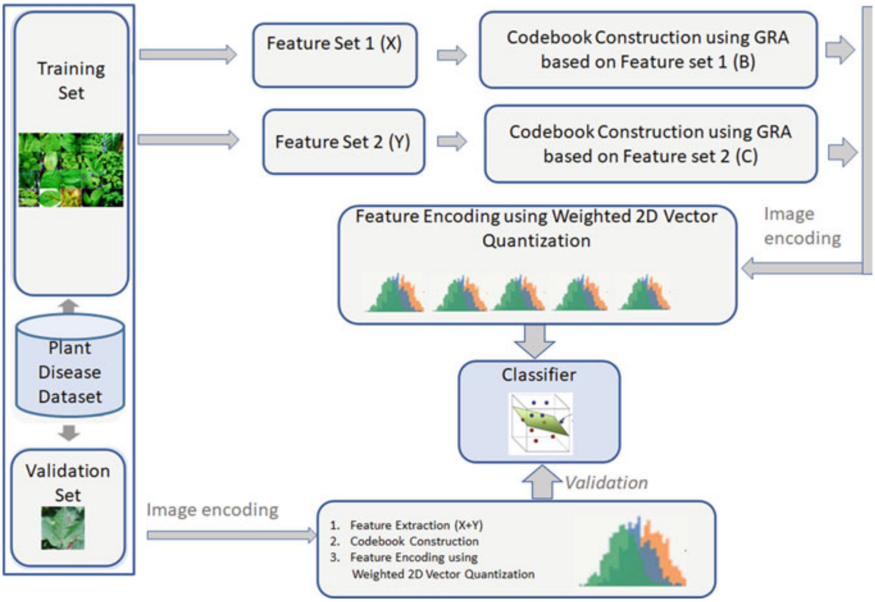


Fig. 1 Proposed diseased plant identification system

and ORB features extractors are used. Based on these methods, two different feature descriptors (X and Y) are generated with different properties.

$$X = \{x_1, x_2, \ldots, x_N\} \in \mathbb{R}^{D_1 \times N} \tag{1}$$

$$Y = \{y_1, y_2, \ldots, y_M\} \in \mathbb{R}^{D_2 \times M} \tag{2}$$

where x_i and y_j are the ith and jth features, generated by considered feature extraction methods with D_1 and D_2 dimensions, respectively. As the plant images are complex in nature, the resulting feature or descriptor vectors are large in size and of high dimensional. Therefore, there is a requirement to find a more representative feature vector, which is performed in the codebook construction phase.

2.2 Codebook Construction

In this phase, clustering is performed on extracted features to find promising and representative features known as visual words. In standard BOF, K-means is used for clustering the feature vectors. But due to the limitations of K-means algorithm, gray relational analysis (GRA)-based clustering method is used to generate the visual words [19]. The GRA-based clustering method is computationally efficient and generates most representative visual words. For better classification results, two different codebooks are generated by performing the clustering on X and Y feature descriptors. The process of clustering algorithm using GRA is given in Algorithm 1. This algorithm is run two times over both set of feature vectors (X and Y) [19], respectively. Let the resulting codebooks generated by the algorithm are $\mathcal{B}$ and $\mathcal{C}$, respectively, from the extracted features.

$$\mathcal{B} = \{b_1, b_2, \ldots, b_K\} \in \mathbb{R}^{D_1 \times N} \tag{5}$$

$$\mathcal{C} = \{c_1, c_2, \ldots, c_K\} \in \mathbb{R}^{D_2 \times M} \tag{6}$$

where b_i and c_j are the ith visual words from each feature set. These codebooks are further used to encode each image into histograms.

3.3 Feature Encoding Using W2DVQ Method

Both of the generated codebooks ($\mathcal{B}$ and $\mathcal{C}$) are passed to the feature encoding phase to generate the histograms. In standard BOF, mainly voting-based methods have been used in the literature [61]. In this method, the histogram is generated based on the vote casted by each feature to a particular visual word. However, these histograms only consider single feature to encode the images due to which less information is incor-

---

Algorithm 1 Clustering approach using GRA

Input: A set of feature vector (X or Y) and a cut-off threshold (T)
Output: Reduced set S of selected features
D=[X or Y]
while $(size(D) > 1)$ **do**
  $temp$=mean(D)
  Choose a reference feature vector (R) near to $temp$;
  $D = D - R$
  Measure GRGs (Γ) in D with R using Eq. (7);

$$\Gamma(R, D) = \sum_{i=1}^{d} [\alpha(i) \cdot \gamma(R(i), D(i))] \tag{7}$$

$$\gamma(R(i), D(i)) = \frac{\min \triangle(i) + \xi \max \triangle(i)}{\triangle(i) + \xi \max \triangle(i)}, \tag{8}$$

  where, $\triangle(i) =\mid R(i) - D(i) \mid$, $\xi \in (0, 1]$, and $\alpha(i)$ is the weighting factor
  The values of GRG must always be between 0 and 1
  The first $T\%$ feature vectors near to 1 are deleted from D ;
  Modify vector $S = [S \;\; R]$;
end while
The resulting set S is considered as the cluster centers or visual words.

---

porated in the representation. So, in this paper, the histograms are generated based on two codebooks by weighted two-dimensional vector quantization (W2DVQ) method [9]. In this method, to represent an image, a two-dimensional code (s), of size $K \times K$ where K is the codebook size, is developed by votes of the its features toward both codebooks. The values of the code ($s(i, j)$) are calculated by a weighted function of two different codes ($\phi_1(i)$ and $\phi_2(i)$) using the following equation.

$$\forall_i \ s(i, j)_{i=1,\ldots,K} = \alpha \cdot \phi_1(i) + (1-\alpha) \cdot \phi_2(j), \ \ j = 1, 2, \ldots, K \tag{9}$$

where α is a weighting factor between (0, 1). The values of $\phi_1(i)$ and $\phi_2(j)$ are generated based on $\mathcal{B}$ and $\mathcal{C}$, respectively, and given in Eq. (10) and Eq. (11), respectively.

$$\phi_1(i) = \begin{cases} 1 & \text{if } i = \underset{j}{\operatorname{argmin}}(\| x - \mathcal{B}_j \|_2), \ \ i = 1, 2, \ldots, K \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

$$\phi_2(j) = \begin{cases} 1 & \text{if } j = \underset{l}{\operatorname{argmin}}(\| y - \mathcal{C}_l \|_2), \ \ l = 1, 2, \ldots, K \\ 0 & \text{otherwise} \end{cases} \tag{11}$$

3.4 Classification

Now, each plant image is represented in a two-dimensional code of visual words. These codes are also known as histograms of visual words. Further, for the classification, support vector machine (SVM) is used. The encoded image in terms of histograms along with the labels is passed to the SVM for training. After training, the system is tested on validation dataset. For the same, a random image from validation set is chosen. Then, these images are represented in the histogram of visual words using the proposed approach and passed these histograms to the SVM. Finally, SVM will predict the label of each validation image.

4 Results and Discussion

The efficiency of the proposed BOF-based diseased plant identification method has been evaluated on 5000 images from PlantVillage dataset [56]. The dataset used for analysis comprises 2500 healthy and 2500 unhealthy or diseased leaf images while the original PlantVillage dataset consists of 54,303 healthy and unhealthy leaf images divided into 38 groups by species and disease. Some of the representative images of healthy and unhealthy categories are shown in Figs. 2 and 3.

(a) (b) (c) (d)

Fig. 2 Representative unhealthy plant leaf images of **a** potato, **b** potato, **c** apple, and **d** apple leaves

(a) (b) (c) (d)

Fig. 3 Representative healthy plant leaf images of **a** potato, **b** potato, **c** apple, and **d** apple leaves

Table 4 Results of the BOF method on PlantVillage dataset in terms of confusion matrices

Class	Healthy	Unhealthy	Method
Healthy	0.64	0.26	SIFT
	0.68	0.27	ORB
	0.67	0.29	SURF
	0.74	0.21	SIFT + ORB
	0.74	0.55	SIFT + SURF
	0.78	0.17	ORB + SURF
Inflamed	0.36	0.74	SIFT
	0.32	0.73	ORB
	0.33	0.71	SURF
	0.26	0.79	SIFT + ORB
	0.26	0.45	SIFT + SURF
	0.22	*0.83*	ORB + SURF

To find the better combination of features, the standard BOF method is applied on PlantVillage dataset with different combinations of the feature extraction methods. Table 4 shows the confusion matrix for each considered combination of feature extraction methods. The true positive rate and false negative rate of the ORB and SURF features are higher than other considered combinations of feature extraction methods.

Table 5 Value of performance parameters returned by the considered methods on PlantVillage dataset

Method	Sensitivity	Specificity	Precision	FNR	Accuracy
SIFT	0.70	0.75	0.73	0.34	0.72
ORB	0.68	0.74	0.72	0.36	0.71
SURF	0.67	0.73	0.71	0.37	0.70
(SURF, ORB)	0.82	*0.86*	*0.85*	0.23	*0.84*
(SIFT, SURF)	*0.88*	0.51	0.62	*0.16*	0.68
(SIFT, ORB)	0.72	*0.86*	0.83	0.32	0.79

Table 6 Performance analysis of the proposed BOF method using the confusion matrix on PlantVillage dataset

Class	Healthy	Unhealthy
Healthy	0.93	0.07
Unhealthy	0.09	0.91

Furthermore, for a fair analysis, sensitivity, specificity, precision, false negative rate, and accuracy of all the considered feature extraction methods have been computed in Table 5. In the leaf classification problem, sensitivity shows the rate at which unhealthy images are recognized. From Table 5, it can be visualized that the SURF and SIFT features together attain the sensitivity of 88% which is better than others followed by the SURF and ORB combination features as 78%. Specificity and precision are another important performance criterion, where specificity finds the rate of recognizing the healthy leaf images and precision is the rate of measuring total predicted healthy images by the classifier. For the performance measure specificity and precision, the combination of SIFT and ORB features attains the better results than other considered methods. From the table, it can be observed that the identification rate is higher in case of ORB and SURF features. Therefore, ORB and SURF are used for feature extraction for further analysis of the proposed system.

Now, with the use of ORB and SURF features, the visual words are generated with GRA-based clustering method and these visual words are used to encode each image into the histogram with the W2DVQ method. These histograms along with labels are passed to SVM classification for training. Table 6 presents the confusion matrix of the proposed BOF method for identifying the unhealthy and healthy images of PlantVillage dataset. In the confusion matrix, rows denote the actual labels while the classifiers' predicted labels are shown in columns. It is visualized from the table that the rate of the identifying the diseased or unhealthy images by the proposed BOF method is 91% which is almost similar to the rate of identifying the healthy images.

Moreover, Table 7 shows the comparison of all the considered methods on PlantVillage dataset. It is perceived from the table that the proposed BOF method attains the mean accuracy of *91.98%* accuracy which is the highest among all the

Table 7 Performance comparison of various proposed BOF methods on PlantVillage dataset

S. No.	BOF approach	PlantVillage dataset
1.	ESMO-BOF [1]	78.35
2.	SBBO-BOF [8]	85.19
3.	GRA-BOF [19]	87.47
4.	Weighted-BOF [9]	78.16
5.	Proposed BOF method	*91.98*

considered state-of-the-art methods. This validates that the proposed system outperforms the existing methods for diseased plant identification.

5 Conclusion and Points for Future Work

In this work, a new BOF method is proposed for the diseased plant identification from leaf images. The proposed BOF method considers different feature descriptors at a time to generate an efficient visual word vocabulary. For this, the proposed method first extracts the SURF and ORB features from the training images and develops two codebooks by clustering through gray relational analysis. Further, the generated codebooks are used by the two-dimensional vector quantization method to encode each image. The encoded images are fed to the classifier along with their annotations for training. Once the classifier is trained, it is used to predict the labels of validation images which are passed to the classifier in encoded form without labels. The proposed BOF method has been tested and validated on publicly available leaf image dataset, i.e., PlantVillage. The experiential results show the efficacy of the new image classification method for disease identification.

In future, the efficiency of the proposed method on a multi-class problem can be conducted wherein the different categories of plant diseases are considered. Moreover, the proposed method can be extended to handle large dataset of leaf images in the big data environment.

References

1. Kumar, S., Sharma, B., Sharma, V.K., Sharma, H., Bansal, J.C.: Plant leaf disease identification using exponential spider monkey optimization. Sustain. Comput. Inform. Syst. (2018)
2. Singh, V., Misra, A.K.: Detection of plant leaf diseases using image segmentation and soft computing techniques. Inform. Process. Agric. **4**(1), 41–49 (2017)
3. Wang, Z., Li, H., Zhu, Y., Xu, T.: Review of plant identification based on image processing. Arch. Comput. Methods Eng. **24**(3), 637–654 (2017)

4. Puja, D., Saraswat, M., Arya, K., et al.: Automatic agricultural leaves recognition system. In: Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012), pp. 123–131. Springer (2013)
5. Elavarasan, D., Vincent, D.R., Sharma, V., Zomaya, A.Y., Srinivasan, K.: Forecasting yield by integrating agrarian factors and machine learning models: a survey. Comput. Electron. Agric. **155**, 257–282 (2018)
6. Kaur, S., Pandey, S., Goel, S.: Plants disease identification and classification through leaf images: a survey. Arch. Comput. Methods Eng. **26**(2), 507–530 (2019)
7. Qiang, Z., He, L., Dai, F.: Identification of plant leaf diseases based on inception v3 transfer learning and fine-tuning. In: International Conference on Smart City and Informatization, pp. 118–127. Springer (2019)
8. Pal, R., Saraswat, M.: Histopathological image classification using enhanced bag-of-feature with spiral biogeography-based optimization. Appl. Intell. **49**(9), 3406–3424 (2019). https://doi.org/10.1007/s10489-019-01460-1
9. Pal, R., Saraswat, M.: A new weighted two-dimensional vector quantisation encoding method in bag-of-features for histopathological image classification. Int. J. Intell. Inform. Database Syst. **13**(2–4), 150–171 (2020)
10. Lowe, D.G.: Object recognition from local scale-invariant features. In: Proceedings of the Seventh IEEE International Conference on Computer Vision, vol. 2, pp. 1150–1157. IEEE (1999)
11. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features. Comput. Vis. Image Understand. **110**, 346–359 (2008)
12. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: Orb: An efficient alternative to sift or surf. In: Proceedings of IEEE International Conference on Computer Vision Workshops, pp. 2564–2571. Barcelona, Spain (2011)
13. Mittal, H., Saraswat, M., Pal, R.: Histopathological image classification by optimized neural network using IGSA. In: International Conference on Distributed Computing and Internet Technology, pp. 429–436. Springer (2020)
14. Bhatia, Y., Bajpayee, A., Raghuvanshi, D., Mittal, H.: Image captioning using Google's inception-resnet-v2 and recurrent neural network. In: 2019 Twelfth International Conference on Contemporary Computing (IC3), pp. 1–6. IEEE (2019)
15. Sivic, J., Zisserman, A.: Video google: a text retrieval approach to object matching in videos. In: IEEE International Conference on Computer Vision, pp. 1470–1478. France (2003)
16. Pal, R., Saraswat, M.: A new bag-of-features method using biogeography-based optimization for categorization of histology images. Int. J. Inform. Syst. Manage. Sci. **1**, 1–6 (2018)
17. Csurka, G., Dance, C., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: Proceedings of Workshop on Statistical Learning in Computer Vision, pp. 1–2. Prague (2004)
18. Mittal, H., Saraswat, M.: Classification of histopathological images through bag-of-visual-words and gravitational search algorithm. In: Proceedings of Soft Computing for Problem Solving, pp. 231–241. Springer (2019)
19. Pal, R., Saraswat, M.: Grey relational analysis based keypoints selection in bag-of-features for histopathological image classification. Recent Patents Comput. Sci. **12**(4), 260–268 (2019)
20. Li, Y., Wang, S., Tian, Q., Ding, X.: A survey of recent advances in visual feature detection. Neurocomputing **149**, 736–751 (2015)
21. Polakowski, W.E., Cournoyer, D.A., Rogers, S.K., DeSimio, M.P., Ruck, D.W., Hoffmeister, J.W., Raines, R.A.: Computer-aided breast cancer detection and diagnosis of masses using difference of gaussians and derivative-based feature saliency. IEEE Trans. Med. Imag. **16**, 811–819 (1997)
22. Ørting, S.N., Petersen, J., Thomsen, L.H., Wille, M.M., de Bruijne, M.: Detecting emphysema with multiple instance learning. In: Proceedings of International Symposium on Biomedical Imaging, pp. 510–513. Washington, United States (2018)
23. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. Int. J. Comput. Vision **60**, 91–110 (2004)

24. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: binary robust invariant scalable keypoints. In: Proceedings of IEEE International Conference on Computer Vision, pp. 2548–2555. Tokyo, Japan (2011)
25. Alahi, A., Ortiz, R., Vandergheynst, P.: Freak: fast retina keypoint. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 510–517. Rhode Island, England (2012)
26. Xu, Q., Varadarajan, S., Chakrabarti, C., Karam, L.J.: A distributed canny edge detector: algorithm and FPGA implementation. IEEE Trans. Image Process. **23**, 2944–2960 (2014)
27. Cruz-Roa, A.A., Ovalle, J.E.A., Madabhushi, A., Osorio, F.A.G.: A deep learning architecture for image representation, visual interpretability and automated basal-cell carcinoma cancer detection. In: Proceedings of International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 403–410. Nagoya, Japan (2013)
28. Mittal, H., Saraswat, M.: A new fuzzy cluster validity index for hyper-ellipsoid or hyper-spherical shape close clusters with distant centroids. IEEE Trans. Fuzzy Syst. (2020)
29. Kulhari, A., Pandey, A., Pal, R., Mittal, H.: Unsupervised data classification using modified cuckoo search method. In: 2016 Ninth International Conference on Contemporary Computing (IC3), pp. 1–5. IEEE (2016)
30. Pal, R., Saraswat, M.: Data clustering using enhanced biogeography-based optimization. In: 2017 Tenth International Conference on Contemporary Computing (IC3). IEEE (2017). https://doi.org/10.1109/ic3.2017.8284305
31. Mittal, H., Saraswat, M.: CKGSA based fuzzy clustering method for image segmentation of rgb-d images. In: 2018 Eleventh International Conference on Contemporary Computing (IC3), pp. 1–6. IEEE (2018)
32. Pal, R., Saraswat, M.: Improved biogeography-based optimization. Int. J. Adv. Intell. Paradigms (2017) (In Press)
33. Pandey, A.C., Tripathi, A.K., Pal, R., Mittal, H., Saraswat, M.: Spiral salp swarm optimization algorithm. In: 2019 4th International Conference on Information Systems and Computer Networks (ISCON), pp. 722–727. IEEE (2019)
34. Jaiswal, K., Mittal, H., Kukreja, S.: Randomized grey wolf optimizer (rgwo) with randomly weighted coefficients. In: 2017 Tenth International Conference on Contemporary Computing (IC3), pp. 1–3. IEEE (2017)
35. Gupta, R., Pal, R.: Biogeography-based optimization with léVY-flight exploration for combinatorial optimization. In: 2018 8th International Conference on Cloud Computing, Data Science & Engineering (Confluence). IEEE (2018). https://doi.org/10.1109/confluence.2018.8442942
36. Mittal, H., Saraswat, M.: An optimum multi-level image thresholding segmentation using non-local means 2d histogram and exponential kbest gravitational search algorithm. Eng. Appl. Artif. Intell. **71**, 226–235 (2018)
37. Mittal, H., Saraswat, M.: An image segmentation method using logarithmic kbest gravitational search algorithm based superpixel clustering. Evolut. Intell. 1–13 (2018)
38. Mehta, K., Pal, R.: Biogeography based optimization protocol for energy efficient evolutionary algorithm: (BBO: EEEA). In: 2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN). IEEE (2017). https://doi.org/10.1109/ic3tsn.2017.8284492
39. Pal, R., Yadav, S., Karnwal, R., Aarti: EEWC: energy-efficient weighted clustering method based on genetic algorithm for HWSNs. Complex Intell. Syst. **6**(2), 391–400 (2020). https://doi.org/10.1007/s40747-020-00137-4
40. Pandey, A.C., Pal, R., Kulhari, A.: Unsupervised data classification using improved biogeography based optimization. Int. J. Syst. Assurance Eng. Manage. **9**(4), 821–829 (2017). https://doi.org/10.1007/s13198-017-0660-2
41. Mittal, H., Pal, R., Kulhari, A., Saraswat, M.: Chaotic kbest gravitational search algorithm (ckgsa). In: Proceedings of International Conference on Contemporary Computing, pp. 1–6. Noida, India (2016)
42. Julong, D.: Introduction to grey system theory. J. Grey Syst. **1**, 1–24 (1989)

43. Jurie, F., Triggs, B.: Creating efficient codebooks for visual recognition. In: Proceedings of IEEE International Conference on Computer Vision, pp. 604–610. California, United States (2005)
44. Peng, X., Wang, L., Wang, X., Qiao, Y.: Bag of visual words and fusion methods for action recognition: comprehensive study and good practice. Comput. Vis. Image Understand. **150**, 109–125 (2016)
45. Huang, Y., Wu, Z., Wang, L., Tan, T.: Feature coding in image classification: a comprehensive study. IEEE Trans. Pattern Anal. Mach. Intell. **36**, 493–506 (2014)
46. van Gemert, J.C., Veenman, C.J., Smeulders, A.W.M., Geusebroek, J.-M.: Visual word ambiguity. IEEE Trans. Pattern Anal. Mach. Intell. **32**, 1271–1283 (2010). https://doi.org/10.1109/tpami.2009.132
47. Liu, L., Wang, L., Liu, X.: In defense of soft-assignment coding. In: Proceedings of International Conference on Computer Vision, pp. 2486–2493. Tokyo, Japan (2011). https://doi.org/10.1109/iccv.2011.6126534
48. Huang, Y., Huang, K., Yu, Tan, T.: Salient coding for image classification. In: Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition, pp. 1753–1760. Colorado, USA (2011). https://doi.org/10.1109/cvpr.2011.5995682
49. Tropp, J.A., Gilbert, A.C.: Signal recovery from random measurements via orthogonal matching pursuit. IEEE Trans. Inform. Theory **53**, 4655–4666 (2007). https://doi.org/10.1109/tit.2007.909108
50. Yang, J., Yu, K., Gong, Y., Huang, T.S., et al.: Linear spatial pyramid matching using sparse coding for image classification. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, pp. 179–1801. Florida, United States (2009)
51. Yu, K., Zhang, T., Gong, Y.: Nonlinear learning using local coordinate coding. In: Proceedings of Advances in Neural Information Processing Systems, pp. 2223–2231. Vancouver, Canada (2009)
52. Wang, J., Yang, J., Yu, K., Lv, F., Huang, T., Gong, Y.: Locality-constrained linear coding for image classification. In: Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3360–3367. California, United States (2010)
53. Yu, K., Zhang, T.: Improved local coordinate coding using local tangents. In: Proceedings of International Conference on Machine Learning, pp. 1–8. Haifa, Israel (2010)
54. Zhou, X., Yu, K., Zhang, T., Huang, T.S.: Image classification using super-vector coding of local image descriptors. In: Proceedings of European Conference on Computer Vision, pp. 141–154. Crete, Greece (2010)
55. Jegou, H., Perronnin, F., Douze, M., Sanchez, J., Perez, P., Schmid, C.: Aggregating local image descriptors into compact codes. IEEE Trans. Pattern Anal. Mach. Intell. **34**, 1704–1716 (2012). https://doi.org/10.1109/tpami.2011.235
56. Mohanty, S.P., Hughes, D.P., Salathé, M.: Using deep learning for image-based plant disease detection. Frontiers Plant Sci. **7**, 1419 (2016)
57. Raza, S.H., Parry, R.M., Moffitt, R.A., Young, A.N., Wang, M.D.: An analysis of scale and rotation invariance in the bag-of-features method for histopathological image classification. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, pp. 66–74. Toronto, Canada (2011)
58. Wang, C.-W., Chen, H.-C.: Improved image alignment method in application to x-ray images and biological images. Bioinformatics **29**, 1879–1887 (2013)
59. Tuzel, O., Porikli, F., Meer, P.: Region covariance: a fast descriptor for detection and classification. In: Proceedings of European Conference on Computer Vision, pp. 589–600. Graz, Austria (2006)
60. Calonder, M., Lepetit, V., Ozuysal, M., Trzcinski, T., Strecha, C., Fua, P.: Brief: computing a local binary descriptor very fast. IEEE Trans. Pattern Anal. Mach. Intell. **34**, 1281–1298 (2011)
61. Stanciu, S.G., Xu, S., Peng, Q., Yan, J., Stanciu, G.A., Welsch, R.E., So, P.T.C., Csucs, G., Yu, H.: Experimenting liver fibrosis diagnostic by two photon excitation microscopy and bag-of-features image classification. Sci. Rep. **4**, 4636–4656 (2014)