# Classification of Fruit Ripeness Levels using Convolutional Neural Network (CNN) and Graph Neural Network (GNN) Methods

Akmal, Diah Chaerani, *Member, IAENG,* Muhammad Ribhan Hadiyan, Calvin Calfi Montolalu

***Abstract*—This research presented a discussion on the classification of fruit ripeness levels using the neural network method. Various studies have been conducted in the literature to develop models for classifying the ripeness levels of fruit. This research aimed to create a classification model for the ripeness levels of various fruit types using a Convolutional Neural Network (CNN) and a Graph Neural Network (GNN). The discussion included a Systematic Literature Review (SLR) and experiments on the topic. SLR result provided understanding into the optimization model used, the types of fruits observed, and the accuracy level for the ripeness classification. However, the types of fruit used in this research were limited to a single kind of fruit. Classifying the ripeness levels of multiple types of fruit also required suitable pre-processing methods to achieve high accuracy. The experiments in this research process started with collecting image data from different fruits. Subsequently, a pre-processing experiment scheme was created using several pre-processing methods. The classification was then conducted using the CNN method with three models: the simple CNN model, MobileNetV2, and ResNet50. Meanwhile, the classification using the GNN method employed two models: VisionGNN and MobileViG. A comparison of the classification precision results is made based on the confusion matrices obtained from each method. Following this discussion, the optimal value was recommended for further processing.**

***Index Terms*—Fruit ripeness classification, Convolutional Neural Network, Graph Neural Network, Confusion matrix.**

## I. Introduction

THE classification of fruit ripeness using machine learning methods is acquiring significant attention in recent years due to its practical applications in agriculture, supply chain management, and the food industries. The ability to accurately determine the level of ripeness of fruits is crucial for optimizing the harvesting, storage, and distribution processes. Given the increasing demand for efficient classification systems, optimizing these machine learning models is essential for ensuring high accuracy and efficiency in real-world applications.

Convolutional Neural Networks (CNNs) and Graph Neural Networks (GNNs) emerge as powerful tools for image-based classification tasks among various machine learning methods. The method has indicated effectiveness in computer vision applications, particularly in classifying the ripeness levels of fruits. In addition, several studies have shown that CNN outperforms traditional methods such as Support Vector Machine (SVM), in classifying fruit ripeness (see [1]). For instance, the need for a more comprehensive classification system that comprises various types of fruit has become evident while SVM has been used in the classification of pineapple, papaya, and melon ripeness levels as mentioned by [2], [3] and [4]. In [5], a discussion is presented on the detection of oil palm trees and loose fruitless ready-to-harvest fresh fruit bunch prediction via a deep learning method. Similarly, in [6] demonstrated that the green apple detection method is based on an optimized YOLOv5 model in an orchard environment. This process is evident in [7], where a fast and efficient Cavendish banana grade classification is achieved using a random forest classifier with a synthetic minority oversampling method.

This research aims to develop a classification model that determines the ripeness level of various fruits using a CNN. It also aims to investigate the key aspects of applying CNN and GNN to fruit ripeness classification. To achieve this process, several fundamental questions will be addressed. Initially, the research will investigate optimization methods commonly used to enhance the accuracy and efficiency of CNN and GNN models. This research offers a detailed analysis of the methods used in classifying fruit ripeness using CNN and GNN models. Using the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines, this study systematically conducts article selection, data extraction, and critical evaluation of findings. A Systematic Literature Review (SLR) is undertaken, and bibliometric analysis is performed using specialized software tools. Subsequently, the research questions serve as a foundation for presenting and discussing the results. There are several questions in this analysis, including:

1) What type of optimization role is used to classify the model for the ripeness levels of various types of fruit through CNN and GNN?
2) What type of fruits have already become the collected image data for implementing CNN and GNN?
3) What is the level of accuracy for the classification

Manuscript received December 5, 2024; revised September 26, 2025. This research is supported by Universitas Padjadjaran Research Grant under the Academic Leadership Grant (ALG) 2024 entitled *Robust Optimization Model Development for Resources Allocation using Information Systems and Mathematical Computation to support the Jawa Barat Commons Goal* with contract number 1435/UN6.3.1/PT.00/2024.

Akmal is an assistant professor at the Department of Computer Sciences, Faculty of Mathematics and Natural Science, Universitas Padjadjaran, Sumedang, 45363, Indonesia (corresponding author to provide e-mail: akmal@unpad.ac.id).

Diah Chaerani is a professor at the Department of Mathematics, Faculty of Mathematics and Natural Science, Universitas Padjadjaran, Indonesia (email: d.chaerani@unpad.ac.id).

Muhammad Ribhan Hadiyan is a bachelor's graduate from the Department of Mathematics, Faculty of Mathematics and Natural Science, Universitas Padjadjaran, Indonesia (e-mail: muhammad20009@mail.unpad.ac.id).

Calvin Calfi Montolalu is a bachelor's graduate of the Department of Computer Sciences, Faculty of Mathematics and Natural Sciences, Universitas Padjadjaran, Indonesia (e-mail: calvin20001@mail.unpad.ac.id).

results based on the confusion matrix obtained from each method?

The structure of this research is organized as follows. Section II covers the literature review, Section IV reviews the methods, and Section V explains the bibliometric analysis as well as a review of the literature. Additionally, Section VI presents the discussion, and Section VII concludes the research. Table I shows the acronyms used throughout the research for easier reference. The literature review focuses on previous research concerning the use of CNN and GNN in classifying fruit ripeness.

TABLE I: Acronyms used in the research.

| Acronym | Definition |
|---|---|
| CNN | Convolutional Neural Network |
| GNN | Graph Neural Network |
| SVM | Support Vector Machine |
| SLR | Systematic Literature Review |
| PRISMA | Preferred Reporting Items for Systematic Reviews and Meta-Analyses |
| GCN | Graph Convolutional Network |
| YOLOv7 | You Only Look Once version 7 (Real-time object detection model) |
| DCNN | Deep Convolutional Neural Network |
| DenseNet | Densely Connected Convolutional Network |
| FDA | Fundamental Data Augmentation |
| SGD | Stochastic Gradient Descent |
| VGG | Visual Geometry Group |
| mAP | Mean Average Precision |
| ANN | Artificial Neural Network |
| KNN | k-Nearest Neighbors |
| GCNN | Graph Convolutional Neural Network |
| RGB | Red, Green, Blue |
| HSV | Hue, Saturation, Value |
| Adam | Adaptive Moment Estimation |

A total of 11 articles were reviewed to compare the optimization methods and model performance used in the classification of various fruit types (see Table II). This table examines the SLR findings of existing research on CNN, GNN, and fruit ripeness classification, as well as the methods employed for model optimization and accuracy improvement. The articles that to be compared with this research are [8], [9], [10], [11], [12], [13], [14], [15], [16], [17] and [18].

As shown in Table II, while numerous studies have focused on the optimization of CNN-based models, research on GNN and its optimization for fruit ripeness classification remains limited. This research aims to contribute to the field by investigating the performance of both CNN and GNN models, comparing their accuracy, and exploring optimization methods to improve classification outcomes.

One significant area in computer vision and machine learning is the application of classification models for detecting fruit ripeness. This challenge falls within the broader scope of image analysis and deep learning, with applications that are widespread in agriculture, the food industry, and supply chain management. Accurately detecting the ripeness of fruits is crucial for ensuring food quality, reducing waste, and improving the efficiency of supply chains.

As the demand for automated solutions increases, the classification of fruit ripeness has gathered interest due to its practical benefits. This comprises automating tasks that are traditionally performed manually, including sorting, grading, and quality control.

Despite the growing interest in the field, several existing research has predominantly focused on a single type of fruit. This narrow focus has limitations, as the ripeness classification system developed for one fruit might not be practical or generalizable to other fruits. Therefore, there is a pressing need for a more comprehensive ripeness classification system that includes a variety of fruits commonly encountered by the general public. The method can improve the strength and applicability of classification models across diverse agricultural contexts.

An exploration of the use of CNNs incorporated with GNNs for fruit ripeness classification is discussed to address the gap in this analysis. This research aims to implement a hybrid model that can accurately identify ripeness stages by leveraging the strong feature extraction capabilities of CNNs together with the strengths of GNNs in relational learning. The combined method considers both visual attributes and relationships between different fruit characteristics, offering significant potential for improving classification performance, particularly in complex datasets that contain diverse fruit types and various ripeness stages. The following Section II provides a brief review of CNN, GNN, and GCN theories.

## II. Material and Methods

### A. Convolutional Neural Network (CNN)

CNN is referred to as a specialized type of artificial neural network designed to process data in the form of images or other multidimensional grids. This model includes spatial data used in pattern recognition and computer vision. CNN has achieved significant popularity and success in various visual pattern recognition tasks, including object detection, image classification, and image segmentation, as demonstrated in [19]. Following the discussion, the architecture of a CNN consisted of multiple layers, including Convolutional, Pooling, and Fully Connected Layers, respectively. The CNN architecture consisted of several convolutional layers followed by stacked pooling layers, ultimately leading to several fully connected layers.

### B. Graph Neural Network (GNN)

Graph Neural Networks (GNNs) represent a subset of machine learning models specifically designed to handle and learn from data structured as graphs. The graphs served as mathematical models that represented nodes connected by edges, making the models ideal for describing relationships and dependencies in complex systems. GNN was applied across various domains, including social networks, recommendation systems, intrusion detection systems, natural language processing, predictive analytics, and computer vision for tasks such as image generation and classification, as seen in [20].

Message passing was a fundamental aspect of GNNs, enabling nodes in a graph to share information iteratively

TABLE II: Summary of key aspects covered in existing articles related to fruit ripeness classification.

| Article | CNN | GNN | Optimization | Fruits | Accuracy |
|---|---|---|---|---|---|
| [8] | ✓ | - | - | ✓ | ✓ |
| [9] | ✓ | - | - | ✓ | ✓ |
| [10] | ✓ | - | - | ✓ | ✓ |
| [11] | ✓ | - | - | ✓ | ✓ |
| [12] | ✓ | - | - | ✓ | ✓ |
| [13] | ✓ | - | ✓ | ✓ | ✓ |
| [14] | ✓ | - | - | ✓ | ✓ |
| [15] | ✓ | - | - | ✓ | ✓ |
| [16] | ✓ | - | - | ✓ | - |
| [17] | ✓ | - | - | ✓ | ✓ |
| [18] | - | - | - | ✓ | - |
| **Our Article** | ✓ | ✓ | ✓ | ✓ | ✓ |

with their neighboring nodes. This process allowed the incorporation of both structural data and node features from the graph. For example, a social circle where individuals hold specific information, such as personal traits. Research could provide a deeper understanding of an individual's characteristics by examining the traits of their friends. In the context of message passing, an individual could be viewed as a node in a graph, with each node representing the traits through node features. The objective of passing the message was to enhance the information of each node by incorporating input from its neighbors (akin to friends) and the neighbors of those neighbors (similar to friends of friends), respectively. This method enabled GNN to effectively capture essential information and relationships in the graph through the use of node features, as seen in [21].

### *C. Graph Convolutional Network (GCN)*

In the context of this research, the Graph Convolutional Network (GCN) is utilized as a specific variant of the Graph Neural Network (GNN) framework. The convolution operation in GCN was analogous to the complication process in CNN. Additionally, input neurons in CNN are multiplied by weight values, commonly referred to as kernels or filters. These filters acted as sliding windows over an input image, enabling the CNN to learn information from neighboring pixels. For instance, when CNN identified an image of a cat, the same filters were applied across the entire input image to detect the presence of a cat in [21]. GCN, as a tool, could collect node information from its surroundings in a convolutional manner. This process was achieved by iteratively aggregating information from neighboring nodes.

The graph convolutional method has been applied in various computer vision tasks, including point cloud classification, scene graph generation, and action recognition. Regarding the discussion, point clouds are a set of three-dimensional points in space, typically captured by LiDAR (Light Detection and Ranging). Scene graph generation transformed the input image into graphs with objects and inter-relationships, often solved by combining object detection with GCN. Action recognition could also be effectively addressed with GCN by converting human skeletons into graphs. However, GCN was powerful in cases where data naturally formed a graph structure, as in the examples mentioned in [21].

### *D. Confusion Matrix (CM)*

In machine learning, a visual evaluation tool called the Confusion Matrix exists. The columns in the confusion matrix represented the predicted class results, while the rows in the confusion matrix signified the actual class results. A Confusion Matrix is calculated by counting all cases in a classification problem. An example of a binary classification problem with a $2 \times 2$ matrix dimension. A confusion matrix can define several measures of algorithm performance, such as the level of examination of positive regions and negative classes. These measures can generally be applied to all classification algorithms as mentioned in [22]. There were four parts in the confusion matrix, namely:

1) True Positive (TP). A condition where the prediction results showed a positive value when the label given was positive.
2) True Negative (TN). A condition where the prediction results showed a negative value when the label given was negative.
3) False Positive (FP). A condition where the prediction results showed a positive value when the label given was negative.
4) False Negative (FN). A condition where the prediction results showed a negative value when the label given was positive.

### *E. PRISMA Methods*

The literature review in this research was conducted in accordance with the PRISMA guidelines. Following this, a bibliometric analysis was performed on the data obtained through the PRISMA process, utilizing RStudio software for assistance.

*1) Search Strategy and Selection Criteria:* During this research, four databases were utilized for article retrieval: Scopus, ScienceDirect, Dimensions, and Google Scholar. The keywords used for article search were organized based on the topics, as shown in Table III. Moreover, some keywords were combined with others, including E, which was a combination of A and B, while F combined B and C, respectively.

The search was limited to the timeframe of 2020-2024 and filtered based on several criteria, as shown in Table IV. These

TABLE III: Number of publications from four databases with five keyword combinations.

| Code | Keywords |
|---|---|
| A | ”Fruit Ripeness Classification” |
| B | ”Convolutional Neural Network” |
| C | ”Graph Neural Network” |
| D | ”Confusion matrix” |

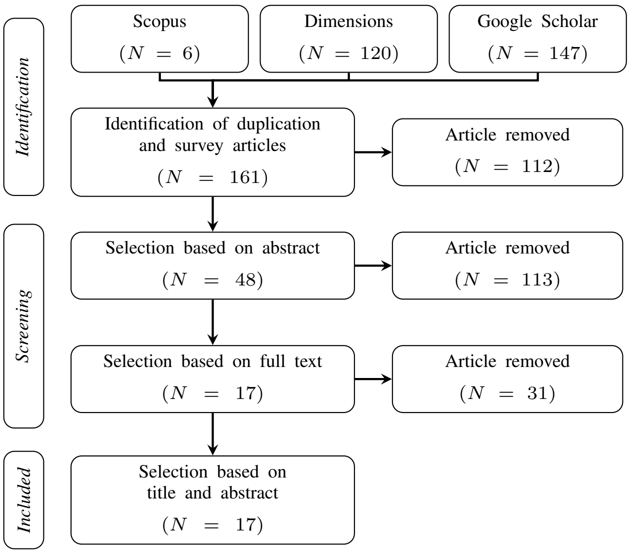


Fig. 1: Selection process based on PRISMA framework.

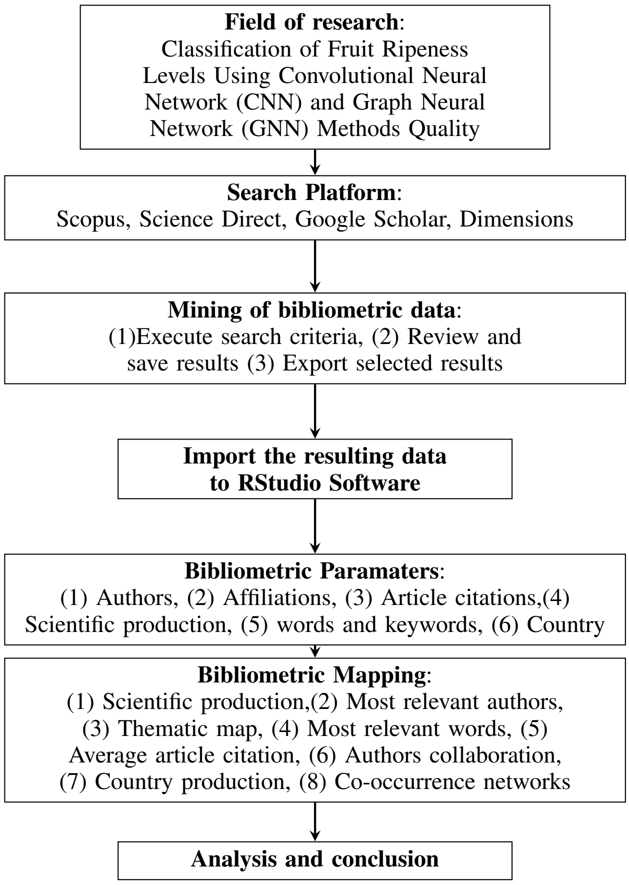


Fig. 2: Bibliometric mapping output steps

criteria included searching through relevant titles, abstracts, and keywords in the databases. Furthermore, data were obtained from articles published in open-access journals, articles were written in English, and the database was researched in mathematics. Based on Table IV, the selection of articles using the keyword combination of A&B was particularly strategic despite producing a relatively moderate number of articles, about 273. The number was optimal for conducting a bibliometric analysis, as it was neither too large nor too small, allowing for a more focused and manageable dataset. Additionally, this set of articles was still sufficiently broad to ensure diversity in the research covered, offering a comprehensive view of the relevant findings. The analysis was comprehensive without being diluted by an excessive volume of less relevant articles, as it focused on this subset.

*2) Selection process based on PRISMA framework:* The articles used were obtained through the PRISMA method, as described in [23] and [24]. This method provided a standardized and accurate framework for arranging the selection criteria, search strategies, data extraction, and analysis procedures. The use of PRISMA improved method precision and results accuracy, serving as a guide for conducting a structured SLR, as can be seen in [25] and [26]. See that Figure 1 shows the selection process stages, where $N$ is the number of references that led to every step of PRISMA.

The PRISMA method began with the initial stage of identification, which involved searching and collecting bibliometric data from various databases. A more comprehensive explanation of the initial stage was provided in Subsection II-E1. According to Table IV, a total of 273 articles were obtained from four databases using two keyword combinations. These articles were further examined in the screening stage, which consisted of two steps.

The first step of the screening stage included checking for duplicates among the identified articles, which appeared in one or more databases with the same title and authors. During the process, the duplication check was performed using a Spreadsheet. After the screening, a total of 95 duplicate articles were removed. A total of 17 review articles were excluded at this stage, in addition to eliminating duplicates, leading to a further reduction of irrelevant entries. Following the process, the remaining articles were then subjected to further analysis. All articles were assessed for relevance to the research topic based on titles and abstracts. A total of 48 articles were identified and referred to as Dataset 1 through this screening stage, which was used in the bibliometric analysis.

The second step was the eligibility stage, during which manual filtering was conducted by reading the full text of the selected articles to evaluate their relevance. From this selection process, a total of 17 articles were identified and referred to as Dataset 2, as shown in Table V.

### *F. Bibliometric Analysis*

A bibliometric analysis was conducted on Dataset 1 using the bibliometrix package in RStudio software to create a scientific data map and provide a comprehensive analysis of the available bibliographic information. The diagram in Figure 2 illustrates the output of bibliometric mapping, based on previous research [27] and in [28].

TABLE IV: The number of publications from databases with keyword combinations.

| Code | Scopus | Science Direct | Dimensions | Google Scholar | Total |
|---|---|---|---|---|---|
| A | 8 | 2 | 188 | 247 | 445 |
| A & B | 6 | 0 | 120 | 147 | 273 |
| A & B & C | 0 | 0 | 2 | 2 | 4 |
| A & B & C & D | 0 | 0 | 1 | 2 | 3 |

TABLE V: Article Selection Results Based on PRISMA Framework.

| Database | Total | Duplication | | Survey | | Abstract/Title | | Full Text | |
|---|---|---|---|---|---|---|---|---|---|
| | | In | Ex | In | Ex | In | Ex | In | Ex |
| Scopus | 6 | 6 | 0 | 6 | 0 | 5 | 1 | 2 | 3 |
| Science Direct | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Dimensions | 120 | 116 | 4 | 104 | 12 | 33 | 71 | 13 | 20 |
| Google Scholar | 147 | 56 | 91 | 51 | 5 | 10 | 41 | 2 | 8 |
| Total | 273 | 178 | 95 | 161 | 17 | 48* | 113 | 17** | 31 |

*Dataset 1 for bibliometric analysis, **Dataset 2 for literature review.

The initial step following the determination of the topic and search platform was the collection of bibliometric data. This phase was conducted concurrently with the article retrieval process in Section II-E1, resulting in the acquisition of Dataset 1, which was obtained from four different databases. To facilitate importing into RStudio, the data format was standardized to match Scopus data. Subsequently, the adjusted data were input into R-Bibliometrix, a package used for bibliometric analysis, as described in [29]. Through this import process, bibliometric parameters were extracted for analysis, including authors, affiliations, article citations, scientific production, countries, words, and keywords. The program further generated bibliometric mapping results based on these parameters.

## III. Results

### A. *Results from Bibliometric Analysis*

The results of the bibliometric analysis of Dataset 1 were discussed in this section. Based on the main information output, a total of 12 articles were identified in Dataset 1. The research was conducted from 2020 to 2024, with an average of 5.95 citations per article. A total of 47 authors were included, with an average of 3.9 co-authors per article.

*1) Evolution of Articles:* An overview of the development of article publication was acquired using R-bibliometrix. From Dataset 1, the number of issues per year was obtained as shown in Table VI and average citations per year were shown in Figure 3. Based on the results, the highest production was in 2023 with a total of twenty articles, while the lowest was in 2020 with one article. Meanwhile, the highest average citation count was in 2021 with six articles.

TABLE VI: Annual Scientific Production in Dataset 1 per year.

| Year | Number of Articles |
|---|---|
| 2020 | 1 |
| 2021 | 4 |
| 2022 | 8 |
| 2023 | 20 |
| 2024 | 15 |

*2) Authors Analysis:* Using R-bibliometrix, which provides the information related to authorship from the article data. The data on article production based on the countries of authors were obtained, where India dominated with a total of 33 articles. Additionally, Figure 4 shows the production over time and the most cited country. This included the author with the most relevant production, specifically in [30], who had four articles published in 2023 and 2024, as shown in Table VII.

*3) Co-occurrence Network Matrix:* Based on Figure 2, a bibliometric mapping that R-bibliometrix generated was a co-occurrence network, which describes the word clusters and their connection to other clusters. In Table tab-clusters, the word clusters and their connection to other clusters are presented.

Table VIII showed the co-occurrence network matrix of terms presented in the title, abstract, and keywords of the articles in Dataset 1. This network revealed clusters of topics formed by terms with similar characteristics. The size of each box for the terms presented in Table VIII showed the importance of words in the context of the articles. In this case, ”deep learning” and ”classification” were the most significant terms, indicating the importance of the research topic in Dataset 1. These two terms were interrelated and situated in the same cluster, signifying frequent discussion of research topics. In the co-occurrence network matrix, terms related to GNN were not found. This indicated that research on the subject was still limited concerning the association with classification and fruit ripeness.

### B. *Result from Systematic Literature Review*

*1) Words Appearance Analysis:* The analysis focused on identifying the most frequent words and creating a thematic map, which included the following key findings. First, the 10 most relevant words appearing in the articles in Dataset 1 were shown in Table IX. ”Deep learning” ranked first with 15 occurrences, followed by ”Classification” and ”Computer Vision” with five occurrences each.

Second, the thematic map matrix, based on density and centrality, displays different clusters of topics organized

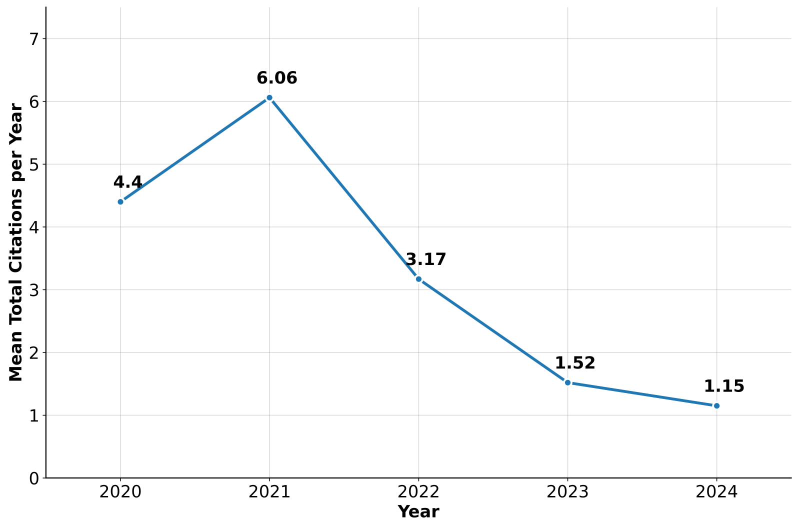


Fig. 3: Average article citations per year.

TABLE VII: Authors Analysis in Dataset 1

| Most Relevant Author | Number of Documents | Authors Production (2020-2024) | |
|---|---|---|---|
| | | Number of Articles | Total Citations per Year |
| Wang Y in [30] | 4 | 4 | 14 |
| Chen W in [30] | 2 | 2 | 11 |
| Li X in [30] | 2 | 2 | 11 |
| Liu M in [30] | 2 | 2 | 11 |
| Liu Y in [31] | 2 | 2 | 10 |
| Saragih R in [32] | 2 | 2 | 26 |
| Suharjito S in [33] | 2 | 2 | 53 |
| Tiwari R in [34] | 2 | 2 | 0 |
| Wang X in [35] | 2 | 2 | 6 |
| Cai J in [36] | 1 | 1 | 3 |

into four themes according to the degree of development (density) and relevance (centrality), as shown in Table X. Clusters such as maturity classification, fine-grained ripeness, and YOLOv7 were highly developed and central to ongoing research in the motor themes. These topics were crucial for further exploration due to the strong interconnections and relevance to other research areas. The niche themes featured topics such as banana and quality detection, which were well-developed but had limited connections to broader research. These niche themes had strong internal development but lacked relevance outside the specific domain. The developing or declining themes included clusters such as multipliers, systematic networks, and fruit ripeness classification. This indicated that these topics were developing or losing traction, showing low development and relevance. Third, the basic themes comprised foundational issues such as deep learning, computer vision, and classification, which were highly relevant and interconnected with other themes, but lacked the internal cohesion of more specific research topics. This analysis showed maturity classification and YOLOv7 as essential research areas, while banana quality detection was a well-developed niche. Developing themes such as fruit ripeness classification required more attention, and the foundational regions, including deep learning and computer vision, continued to play a crucial role in the field. The observations from Dataset 2 were presented as SLR results in this section. Dataset 2 comprised 17 articles that had undergone the whole selection process and were published between 2020 and 2024. A review of the topics in the 17 articles and the optimization method was shown in Tables XI and XII.

## IV. Discussion

### A. *State-of-the-art Classification of Fruit Ripeness Levels Using CNN and GNN Methods*

The answers to the research questions stated in Section II were discussed in this section. The state-of-the-art Classification of fruit ripeness levels using CNN and GNN methods revealed research trends, playing a significant role in identifying new topics. This section also discussed the application of optimization models in research related to the topic. The state-of-the-art Classification of Fruit Ripeness Levels Using CNN and GNN methods is shown in Table XI and Table XII. These tables displayed the covered topics in each article, along with explanations of the objectives and constraints used in the optimization models.

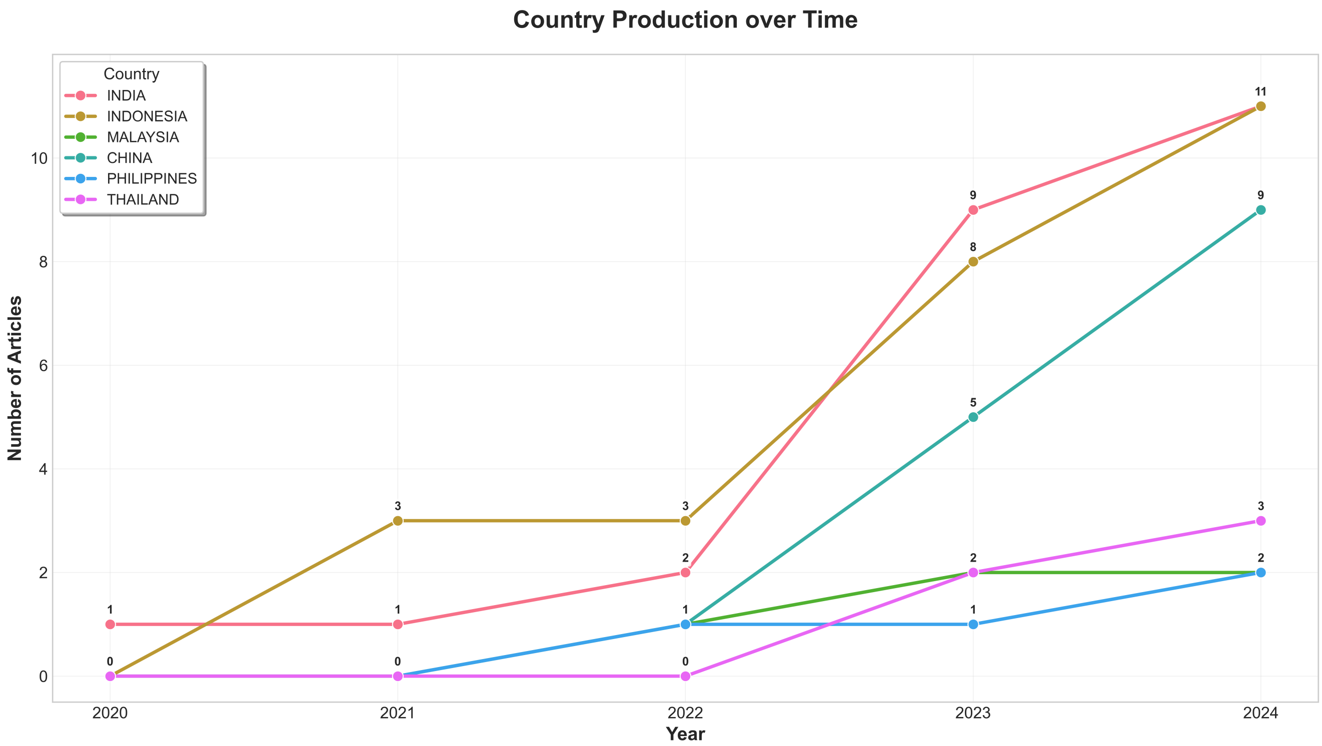


Fig. 4: Country production over time of Dataset 1.

TABLE VIII: Co-occurrence Network Matrix of Dataset 1.

| Main Cluster | Related Terms | Connections to Other Clusters |
|---|---|---|
| Deep Learning | Oil Palm | Maturity, Machine Learning, Graph Convolutional Neural Network, Computer Vision, Agricultural Image Analysis, Convolutional Neural Network, Ripeness |
| Maturity | YOLOv7, ResNet50 | Deep Learning, Convolutional Neural Network, Ripeness |
| Ripeness | Object Detection, YOLO, SVM, YOLOv4 | Deep Learning, Maturity, Agricultural Image Analysis, Convolutional Neural Network |
| Agricultural Image Analysis | Agricultural Technology, Support Vector Machines | Deep Learning, Ripeness, Machine Learning |
| Machine Learning | Banana Life Classification, Mask R-CNN, Visual Object Detection, One-Stage Model, Attention Mechanism, Fine-Grained, Maturity Classification | Deep Learning, Computer Vision, Agricultural Image Analysis, Graph Convolutional Neural Network |
| Computer Vision | DeepLabv3+, Fruit Maturity | Deep Learning, Machine Learning, Image Processing |
| Convolutional Neural Network | Fruits, Gabor Filters, Deep Neural Networks, Avocado Ripeness Classification, YOLOv8, CBAM, Fruit Ripeness Detection, Artificial Olfactory Sensor | Deep Learning, Maturity, Ripeness |
| Graph Convolutional Neural Network | Banana, Quality Detection, Agriculture Management, Fuji Apples | Deep Learning, Machine Learning, Agricultural Image Analysis |
| Image Processing | Bunch of Bananas | Computer Vision |

### *B. Research Trends and Gaps on Classification of Fruit Ripeness Levels Using CNN and GNN Methods*

Based on bibliometric analysis, there has been continuous growth in related publications, as evidenced by the significant increase from 2022 to 2023. According to Table IX, all keywords exhibited an upward trend, particularly those related to deep learning, indicating significant advancements in the field of fruit ripeness classification. The co-occurrence network matrix in Table VIII showed the relationships between terms found in articles from Dataset 1. Moreover, keywords related to GNN were absent in the co-occurrence network matrix, indicating that this topic was still

TABLE IX: Most relevant words of Dataset 1.

| Most relevant words | Occurrences |
|---|---|
| Deep learning | 15 |
| Classification | 5 |
| Computer Vision | 5 |
| CNN | 4 |
| Convolutional Neural Network | 4 |
| Machine Learning | 4 |
| Ripeness | 4 |
| YOLOv7 | 4 |
| Attention Mechanism | 3 |
| Banana Ripeness | 3 |

underrepresented or unused, presenting a promising area for future research.

### *C. How Optimization models have been applied in research related to the Classification of Fruit Ripeness Levels Using CNN and GNN Methods*

Based on the results shown in Table XII, the articles in Dataset 2 primarily focused on optimization challenges in classifying fruit ripeness levels using various methods, including CNN and GNN. The majority of this research utilized advanced deep-learning architectures to enhance the accuracy of ripeness classification across various fruit types. In particular, several articles employed CNN-based methods, incorporating multiple optimization techniques to improve model performance. For instance, [35] applied Gas Chromatography-Mass Spectrometry (GC-MS) and Deep Convolutional Neural Network (DCNN) combined with Densely Connected Convolutional Networks (DenseNet) to classify fruit ripeness. Similarly, [37] leveraged the Bayesian Optimization Algorithm alongside multiple CNN architectures, such as GoogleNet and ResNet18, indicating the importance of optimization in model training and performance tuning.

The use of data augmentation strategies, as observed in articles [39] and [42], highlights the critical role of optimizing dataset preparation in improving the strength of CNN models. Methods such as grayscaling and sharpening were used to preprocess the image, thereby improving the models' effectiveness in identifying ripeness levels. During the research, the inclusion of GNN in this domain proved to be significant. For example, [40] implemented Knowledge-Embedded Graph Convolutional Neural Networks (KEGCNNs) and Graph Convolutional Neural Networks (GCNNs) to address classification challenges, indicating a shift towards leveraging graph-based methods for fruit classification tasks.

The optimization models discussed in these articles predominantly improved classification accuracy while minimizing computational costs. For instance, [43] compared the performance of CNN with feature-based classifiers and penalized methods, demonstrating the importance of model selection and optimization strategies in achieving superior classification outcomes.

### *D. Results Experiment Convolutional Graph Network*

During this research, the objects of focus included images of Apples, Oranges, Limes, Guavas, Bananas, and Pomegranates. Each of the six classes of fruit images had two subclasses, namely good and bad quality. During the analysis, the fruit image data was obtained from a site called Kaggle.org. The data provided in the dataset was divided into fruit type, fruit quality, and combined image data. Fruit type and quality image data were split into two, namely training data and testing data. Additionally, training data was utilized for the model training process. Testing and combined image data were used for the evaluation process of the model that was formed.

*1) Datasets:* This stage prepared the dataset used to measure the model's accuracy. In creating the model, the fruit image was divided into two classes, namely good and rotten fruit. Each class had six subclasses of fruit image, namely Apple, Orange, Lime, Guava, Banana, and Pomegranate. The data collection process resulted in a dataset comprising a total of 19,526 fruit images. Each image in the dataset had a size of 256 $\times$256 pixels, captured from various angles, including the front, back, top, bottom, and a 180-degree rotation. Following this discussion, the dataset can be accessed on the Kaggle.org website.

There was a division of the number of fruit images, which were divided into two qualities, namely good and bad, in Table XIII. The number of pictures for each fruit consisted of $\pm$1000 of both good and bad quality.

After the dataset was collected, the data was divided into two, namely training data and testing data. The machine used the training data to train the model, with a ratio of 70%. On the other hand, the machine used testing data to evaluate the trained model at a ratio of 30%.

*2) Pre-Processing Data:* At this stage, the dataset that was divided was processed to continue to the model creation stage. This was performed to enable the classified image to be used properly. During the process, five types of pre-processing were studied with various values. The list of pre-processing steps, along with the values studied, is shown in Table XIV.

*3) Model Development:* During this stage, training was conducted on the training dataset that was divided, and the training results were validated. The training process was repeated with a total of 10 epochs and the AdamW optimizer to determine the learning rate value, see [52]. In this research, three types of CNN model architectures were compared: simple CNN, MobileNetV2, and ResNet50. The first was a Simple CNN Model architecture, which was shown in Table XV. Within this architecture, there are two types of convolutional layers: Conv2D and Conv2D_1. Each layer was followed by a pooling layer, specifically MaxPooling2D and MaxPooling2D_1. Additionally, each layer concluded with three fully connected layers: Flatten, Dense, and Dense_1, respectively.

The second architecture used was the MobileNetV2 Model architecture, as shown in Table XVI. In this model, a pre-trained model was used in the first layer, followed by two types of fully connected layers.

The third architecture that was used in this research was the ResNet50 Model architecture, as shown in Table XVII. In this architecture, there was one type of pre-trained model

TABLE X: Thematic map matrix based on density and centrality

| | | **Centrality** | |
|---|---|---|---|
| | | **Low** | **High** |
| **Density** | **High** | **Niche Themes**<br>• CNN, Machine Learning, SVM<br>• Attention mechanism, maturity classification, Banana | **Motor Themes**<br>• Ripeness, Yolov7, Multiplier<br>• Deep Convolutional Neural Network, Fruit Ripeness Classification |
| | **Low** | **Emerging or Declining Themes**<br>• Convolutional Neural Network, Fruit Ripeness Detection, Yolov8.<br>• Banana Ripeness, Image Processing, Yolo. | **Basic Themes**<br>• Mask R-CNN, Strawberry, Convolutional Neural Network.<br>• Deep Learning Classification, Computer Vision. |

TABLE XI: Covered ripeness problem topics in Dataset 2.

| Article | Topics on Fruit Ripeness |
|---|---|
| [35] | Portable Fruit Ripeness Prediction System for Mango, Peach, and Banana. |
| [37] | Classification of Strawberry Fruit Ripeness and Quality. |
| [38] | Banana Ripeness Recognition. |
| [39] | Determining Mangrove Fruit Ripeness. |
| [40] | Banana Fruit Ripeness Detection for Quality Grading in Smart Farming. |
| [41] | Fine-Grained Maturity Classification of Fuji Apples in Open-World Orchard Environments. |
| [42] | Automated Fruit Ripeness and Quality Assessment: A Case Study on Oranges, Papaya, and Banana. |
| [43] | Strawberry Ripeness Detection Using Image Data. |
| [32] | Banana Ripeness Classification. |
| [44] | Oil Palm Fresh Fruit Bunch Ripeness Classification On Mobile Devices. |
| [45] | Automated Classification of Strawberry Ripeness. |
| [46] | Ambon Banana Maturity Classification. |
| [47] | Avocado Ripeness Classification. |
| [48] | Fruit Ripeness Analysis Of Sugar Apples (Annona Squamosa). |
| [49] | Non-Destructive Maturity Grading of Pineapples. |
| [50] | Estimating The Ripening State of Fuji Apples. |
| [51] | Predicting Banana Fruit Ripeness and Quality. |

in the first layer, followed by two types of fully connected layers. In the data pre-processing phase of GNN, image formats were converted to a uniform format, specifically JPG. This was necessary because the dataset contained some images in PNG format, which had different dimensions compared to the JPG images. Uniformity in format was crucial for maintaining consistency in the input image dimensions.

The next step in the data pre-processing phase involved resizing the image to achieve uniform dimensions, followed by image augmentation on the training dataset to increase its variability, and concluding with the conversion of the image to a tensor data type. The parameters used in the data pre-processing stage were based on some of the parameters used by the VisionGNN model developed by [53].

Table XVIII shows the detailed parameter values for each data pre-processing step used. During this stage, the model was trained using data that had undergone pre-processing. The model was trained on the yoga pose image dataset located in the training folder. This stage was crucial for the model to learn the features necessary for the accurate classification of yoga poses.

Table XIX showed the training parameters used during the research. In addition, AdamW optimizer (see [52]) was used with an initial learning rate of 0.002. A cosine annealing schedule was implemented for the learning rate scheduler. The loss function used was cross entropy with a label smoothing value of 0.1. Following the discussion, the model was trained for 50 epochs during the process. The loss value and accuracy were tracked for each epoch to monitor the training process.

*4) Preprocessing:* During this research, several combinations of pre-processing methods were applied to the training data and test data. Several methods were combined, including color change and brightness level

TABLE XII: Used optimization methods in Dataset 2.

| Article | Method in Fruit Ripeness |
|---|---|
| [35] | Gas ChromatographyMass Spectrometry (GC-MS); Colorimetric Sensing Combinatorics; Deep Convolutional Neural Networks (DCNN); Densely Connected Convolutional Network (DenseNet). |
| [37] | Fundamental Data Augmentation (FDA); Learning-to-Augment Strategy (LAS); Bayesian Optimization Algorithm; GoogleNet; ResNet18; ShuffleNet. |
| [38] | ResNet 34; ResNet 101; VGG 16; VGG 19. |
| [39] | Grayscaling; Adaptive Threshold; Sharpening; Smoothing. |
| [40] | Knowledge Embedded-Graph Convolutional Neural Networks (KEGCNNs); Graph Convolution Neural Networks (GCNNs). |
| [41] | CNN-based fine-grained lightweight architecture; attention mechanism; lightweight structure. |
| [42] | Image resizer; Stochastic gradient descent (SGD) algorithm; Convolutional neural network (CNN); Nearest neighbor interpolation algorithm; Visual geometry group with 16 layers (VGG). |
| [43] | Convolutional Neural Networks; Feature-based Classifiers; Penalized Multinomial Regression; Penalized Discriminant Analysis. |
| [32] | Convolutional Neural Network; MobileNet V2; NASNetMobile; Computer Vision; Machine Learning. |
| [44] | Oil lightweight Convolutional Neural Network (CNN); ImageNet transfer learning; 9-angle crop data augmentation; post-training quantization; Float16 quantization; TensorFlow Lite implementation. |
| [45] | Convolutional neural network (CNN); feature extraction; surface color analysis; size and shape analysis. |
| [46] | Convolutional Neural Network (CNN); Image enhancement features; Image processing using MATLAB R2022a software. |
| [47] | Multi-Channel Hybrid Deep Neural Networks (MCHDNN); Visual Geometry Group 16 (VGG16); EfficientNetB0 architectures; early fusion-based feature concatenation; Convolutional Neural Network (CNN). |
| [48] | Proposed deep learning model (ECD-DeepLabv3+); lightweight backbone (MobileNetV2); efficient channel attention (ECA) module; Dense ASPP module; coordinate attention (CA) module; validation using a self-made dataset of harvested optical images. |
| [49] | Non-destructive approach for pineapple maturity grading; object segmentation framework; multi-object sampling technique in augmentation process; model training with small dataset; robustness evaluation of Mask R-CNN compared to other models; analysis of mean average precision (mAP) and detection ratio; optimal threshold selection for performance enhancement; and detailed experimental results analysis for future insights. |
| [50] | Convolutional neural network (CNN) model; hyperspectral information in the visible and near-infrared (Vis/NIR) regions was analyzed; the classification performance of the CNN was compared with three alternative methods: artificial neural networks (ANN), support vector machines (SVM), and k-nearest neighbors (KNN). |
| [51] | Image collection; Dataset creation; Image augmentation; Convolutional neural networks (CNN); AlexNet technique; Model training and evaluation; Accuracy measurement. |

TABLE XIII: Distribution of the number of fruit images of each quality.

| Fruit Names | Good Quality | Bad Quality |
|---|---|---|
| Apple | 1,149 | 1,141 |
| Banana | 1,113 | 1,087 |
| Guava | 1,152 | 1,129 |
| Lime | 1,094 | 1,085 |
| Orange | 1,216 | 1,159 |
| Pomegranate | 5,940 | 1,187 |

TABLE XIV: Pre-processing List

| Preprocessing | Value |
|---|---|
| Color | RGB, Grayscale, YIQ, YUV, and HSV |
| Central Crop | 50% - 70% |
| Brightness | 90% - 120% |
| Contrast | 70% - 90% |
| Gamma Light | 110% - 140% |
| Hue | 20% - 50% |
| Saturation | 60% - 80% |

adjustment methods in the image. The results of testing the combination of these pre-processing methods were evaluated from the outcomes of model training.

Based on the experimental results of each pre-processing method implemented, the findings showed that the most effective pre-processing process was a combination of red-green-blue (RGB) coloring methods, with a central crop of 70%, a brightness level of 110%, and a Gamma light of 110%.

1) Simple CNN. The first architecture used in this research was a simple CNN architecture. This model was constructed using seven layers, and the number of

TABLE XV: Simple CNN model architecture

| Layer | Type | Output Shape | Param |
|---|---|---|---|
| Conv2D | Conv2D | 240×240 × 32 | 896 |
| MaxPooling2D | MaxPooling2D | 120×120 × 32 | 0 |
| Conv2D_1 | Conv2D | 120×120 × 64 | 18,496 |
| MaxPooling2D_1 | MaxPooling2D | 60×60 × 64 | 0 |
| Flatten | Flatten | 230,400 | 0 |
| Dense | Dense | 128 | 29,491,328 |
| Dense_1 | Dense | 12 | 1,548 |

TABLE XVI: MOBILENETV2 model architecture

| Layer | Type | Output Shape | Param |
|---|---|---|---|
| MobileNetV2 | Functional | 1,280 | 2,257,984 |
| Dense | Dense | 128 | 1,633,968 |
| Dense_1 | Dense | 12 | 1,548 |

TABLE XVII: RESNET50 model architecture

| Layer | Type | Output Shape | Param |
|---|---|---|---|
| ResNet50 | Functional | 2,048 | 23,587,712 |
| Dense | Dense | 128 | 262,272 |
| Dense_1 | Dense | 12 | 1,548 |

TABLE XVIII: Data Preprocessing Parameter

| Stage | Details |
|---|---|
| Image Resize | Image Resize 224× 224 |
| Image Augmentation | Random Erasing 0.25 |
| | Random Augment TRUE |
| | Normalize Mean = 0.485, 0.456, 0.406 |
| | Std = 0.229, 0.224, 0.225 |
| Data Loader | Batch Size 128 |

TABLE XIX: Training Parameters

| Parameter | Method |
|---|---|
| Optimizer | AdamW |
| Loss Function | Cross Entropy |
| Learning Rate Scheduler | Cosine Annealing Schedule |

parameters utilized in this model totaled 29,512,268. The architecture of the model used during the analysis is shown in Table XX.

TABLE XX: Simple CNN Architecture

| Layer | Type | Output Shape | Param |
|---|---|---|---|
| Conv2D | Conv2D | 240×240 × 32 | 896 |
| MaxPooling2D | MaxPooling2D | 120×120 × 32 | 0 |
| Conv2D_1 | Conv2D | 120×120 × 64 | 18,496 |
| MaxPooling2D_1 | MaxPooling2D | 60×60 × 64 | 0 |
| Flatten | Flatten | 230,400 | 0 |
| Dense | Dense | 128 | 29,491,328 |
| Dense_1 | Dense | 12 | 1,548 |

The first to fourth layers were convolutional layers that performed feature extraction from the fruit image. The remaining layers were fully connected layers that formed the input from the image into smaller dimensions.

2) MobileNetV2
The second model used in this research was the MobileNetV2 architecture. This model was constructed using three layers, and the number of parameters employed in this model totaled 2,423,500. The architecture of the model used during the research is shown in Table XXI.

TABLE XXI: MobileNetV2 Architecture

| Layer | Type | Output Shape | Param |
|---|---|---|---|
| MobileNetV2 | Functional | 1,280 | 2,257,984 |
| Dense | Dense | 128 | 1,633,968 |
| Dense_1 | Dense | 12 | 1,548 |
| **Total Param** | | | **2,423,500** |

The first layer used in this research process was a pre-trained model. The weight value of the model was obtained from the training results using the ImageNet dataset. The remaining layers were fully connected layers that transformed the image's input into smaller dimensions.

3) ResNet50
The first model used in this research was the ResNet50 architecture. This model was constructed using three layers, and the number of parameters employed in this model totaled 23,851,532. The architecture of the model used during the process is shown in Table XXII. The first layer used in this research was a pre-trained model. The weight value of the model was obtained from the training results using the ImageNet dataset. Additionally, the remaining layers were fully connected layers that formed input from the image into smaller dimensions.

TABLE XXII: ResNet50 Architecture

| Layer | Type | Output Shape | Param |
|---|---|---|---|
| ResNet50 | Functional | 2,048 | 23,587,712 |
| Dense | Dense | 128 | 262,272 |
| Dense_1 | Dense | 12 | 1,548 |
| **Total Param** | | | **23,851,532** |

TABLE XXIII: Model Evaluation Results of CNN

| Metric | Value |
|---|---|
| Accuracy | 83.3% |
| Precision | 84.5% |
| Recall | 82.5% |
| $F_1$ Score | 86.0% |

*5) Model Evaluation:* In each model experiment, training was performed using the following hyperparameters: epoch = 10, batch size = 64, and learning rate = 0.001. Four metrics were considered in the model evaluation: accuracy, loss, precision, recall, and $F_1$-score.

Based on the model experiments and pre-processing results obtained, the architecture suitable for use as a fruit quality classification model was MobileNetV2, with RGB coloring

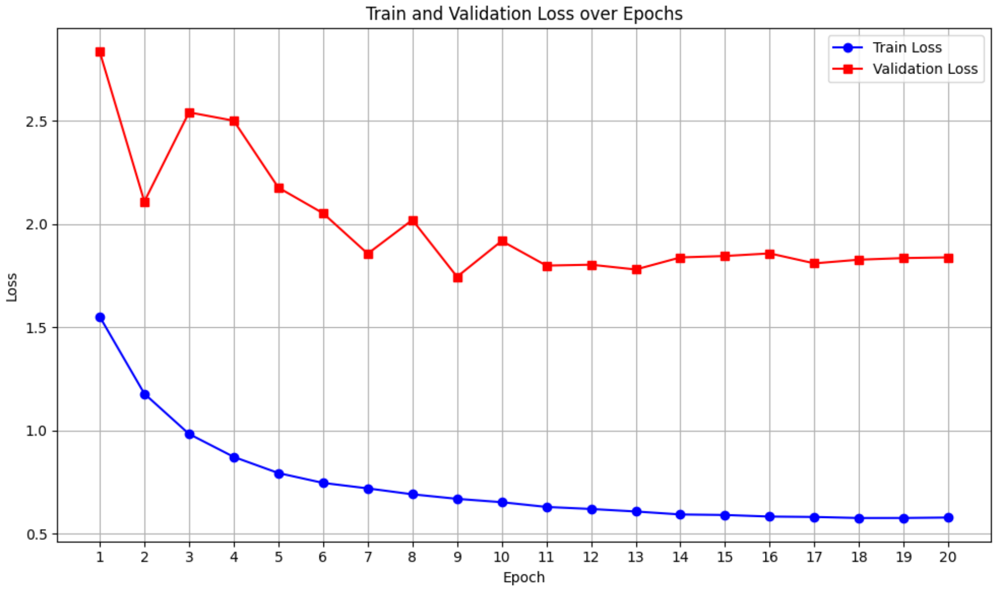


Fig. 5: The Epoch comparison of train loss and validation loss.

and the following pre-processing method: central crop = 70%, brightness level = 110%, and Gamma light = 110%.

After completing the model training, a re-evaluation was conducted using the confusion matrix calculation. The evaluation results of this CNN model during the analysis were shown in Table XXIII. Table XXIV presents the evaluation of training and validation accuracy obtained from the GNN during the study.

TABLE XXIV: Accuracy Evaluation of GNN

| Epoch | Training Loss | Validation Loss | Training Accuracy |
|---|---|---|---|
| 1 | 1.55173 | 2.83615 | 57.71% |
| 2 | 1.17799 | 2.10985 | 74.38% |
| 3 | 0.98279 | 2.54217 | 82.86% |
| 4 | 0.87171 | 2.50164 | 87.18% |
| 5 | 0.79334 | 2.17712 | 91.03% |
| 6 | 0.74553 | 2.05302 | 92.88% |
| 7 | 0.71892 | 1.85694 | 93.96% |
| 8 | 0.69024 | 2.02051 | 94.86% |
| 9 | 0.66822 | 1.74496 | 95.83% |
| 10 | 0.65195 | 1.91865 | 96.45% |
| 11 | 0.62902 | 1.79907 | 96.98% |
| 12 | 0.61915 | 1.80348 | 97.36% |
| 13 | 0.60694 | 1.78018 | 97.98% |
| 14 | 0.59263 | 1.83876 | 98.48% |
| 15 | 0.59001 | 1.84557 | 98.45% |
| 16 | 0.58240 | 1.85824 | 98.69% |
| 17 | 0.58049 | 1.80990 | 98.72% |
| 18 | 0.57538 | 1.82779 | 98.98% |
| 19 | 0.57562 | 1.83560 | 98.88% |
| 20 | 0.57772 | 1.83891 | 98.82% |

Training refers to the process of adjusting the parameters of a neural network to minimize the cost function. Additionally, validation refers to the process of measuring the accuracy of a model on a subset of data. Accuracy was a measure of how well a model predicted the correct output based on the input. The epoch comparison of train loss, validation loss, and train accuracy is shown in Figure 5. There was a notable difference between the training loss and the validation loss. Training demonstrated that the model was learning, while validation showed that it was able to generalize well to new data. It can be observed that the error loss decreased steadily over time, while accuracy increased. This showed that the model was still learning from the training data and had not been over-fitted. When the loss reaches saturation, it indicates that the model has reached its maximum capacity to learn from the training data.

## V. Conclusions

In conclusion, this research successfully explored the classification of fruit ripeness levels using a convolutional neural network (CNN) and a graph neural network (GNN) method. The study includes a systematic literature review and a bibliometric analysis. Based on a bibliometric data search, articles were collected from several databases, including Dimensions, Google Scholar, Scopus, and Science Direct. Subsequently, the selection was conducted using the PRISMA framework, resulting in 48 articles for bibliometric analysis in Dataset 1 and 17 articles used in the state-of-the-art analysis, as presented in Dataset 2.

The results showed that research concerning the use of Classification of Fruit Ripeness Levels Using Convolutional Neural Network (CNN) and Graph Neural Network (GNN) Methods was still relatively underexplored, indicating potential for further development. This observation was substantiated by a bibliometric analysis, which revealed the absence of keywords related to GNN. Several methods will be employed for exploration, including the use of the GNN library and framework, and testing it with popular graph datasets. Relating to the conclusion, comparing the models formed through performance evaluation. The next plan associated with this topic was the research and development of data sets, as well as the classification of local Indonesian fruits.

For the experiment presented, the results showed that error loss decreased steadily over time, while accuracy increased. This indicated that the model was still learning from the training data and had not been over-fitted. When the loss reaches saturation, it suggests that the model has reached its maximum capacity to learn from the training data.

## REFERENCES


[1] K. Sabanci, M. F. Aslan, E. Ropelewska, and M. F. Unlersen, "A convolutional neural networkbased comparative study for pepper seed classification: Analysis of selected deep features with support vector machine," *Journal of Food Process Engineering*, vol. 45, no. 6, p. e13955, 2022.

[2] Y. B. E. Purba, N. F. Saragih, and A. P. Silalahi, "Design of a pineapple ripeness detection tool using a microcontroller using the convolutional neural network (cnn) method," *Jurnal Ilmiah Teknik*, vol. 2, no. 1, pp. 46–51, 2022.

[3] N. Zulkifli, N. Hashim, H. H. Harith, M. F. Mohamad Shukery, and D. I. Onwude, "Prediction of the ripening stages of papayas using discriminant analysis and support vector machine algorithms," *Journal of the Science of Food and Agriculture*, vol. 102, no. 8, pp. 3266–3276, 2022.

[4] S. R. S. Baki, I. M. Yassin, A. H. Hasliza, and A. Zabidi, "Non-destructive classification of watermelon ripeness using mel-frequency cepstrum coefficients and multilayer perceptrons," in *The 2010 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2010, pp. 1–6.

[5] M. M. Daud, Z. Kadim, and H. H. Woon, "Detection of oil palm tree and loose fruitlets for fresh fruit bunch′s ready-to-harvest prediction via deep learning approach," *IAENG International Journal of Computer Science*, vol. 50, no. 4, pp. 1183–1193, 2023.

[6] W. Zhang, Y. Zhao, Y. Guan, T. Zhang, Q. Liu, and W. Jia, "Green apple detection method based on optimized yolov5 under orchard environment," *Engineering Letters*, vol. 31, no. 3, pp. 1104–1113, 2023.

[7] S. Arwatchananukul, R. Saengrayap, S. Chaiwong, and N. Aunsri, "Fast and efficient cavendish banana grade classification using random forest classifier with synthetic minority oversampling technique," *IAENG International Journal of Computer Science*, vol. 49, no. 1, pp. 46–54, 2022.

[8] T. C. Lew Sook Ling, Mohammadmahdi Naghipour, "A review of ai techniques in fruit detection and classification: Analyzing data, features and ai models used in agricultural industry," *International Journal of Technology*, vol. 15, no. 3, pp. 585–596, May 2024.

[9] S. A. Ghazalli, H. Selamat, N. Khamis, and M. F. Haniff, "Short review on palm oil fresh fruit bunches ripeness and classification technique," *Journal of Advanced Research in Applied Mechanics*, vol. 106, no. 1, pp. 37–47, Jul 2023.

[10] P. Baglat, A. Hayat, F. Mendona, A. Gupta, S. S. Mostafa, and F. Morgado-Dias, "Non-destructive banana ripeness detection using shallow and deep learning: A systematic review," *Sensors*, vol. 23, no. 2, p. 738, 2023.

[11] N. Ragu and J. Teo, "Object detection and classification using few-shot learning in smart agriculture: A scoping mini review," *Frontiers in Sustainable Food Systems*, vol. 6, pp. 1–9, 2023.

[12] N. Aherwadi and U. Mittal, "Fruit quality identification using image processing, machine learning, and deep learning: A review," *Advances and Applications in Mathematical Sciences*, vol. 21, no. 5, pp. 2645–2660, 2022.

[13] T. Tamilarasi and P. Muthulakshmi, "Effectiveness of deep feature extraction algorithm in determining the maturity of fruits: A review," *International Journal on Recent and Innovation Trends in Computing and Communication*, vol. 11, no. 7, pp. 105–117, 2023.

[14] J. L. A. Deguzman, A. C. C. Villagomez, and E. R. Arboleda, "Beyond traditional farming: An exploration of machine learning technologies in banana agriculture a literature review," *International Journal of Scientific Research and Engineering Development*, vol. 7, no. 1, p. 515, 2024.

[15] M. Rizzo, M. Marcuzzo, A. Zangari, A. Gasparetto, and A. Albarelli, "Fruit ripeness classification: A survey," *Artificial Intelligence in Agriculture*, vol. 7, pp. 44–57, 2023.

[16] J. W. Lai, H. R. Ramli, L. I. Ismail, and W. Z. Wan Hasan, "Oil palm fresh fruit bunch ripeness detection methods: A systematic review," *Agriculture*, vol. 13, no. 1, p. 156, 2023.

[17] J. Y. Goh, Y. Md Yunos, and M. S. Mohamed Ali, "Fresh fruit bunch ripeness classification methods: A review," *Food and Bioprocess Technology*, vol. 18, no. 1, pp. 183–206, 2025.

[18] S. Yu, X. Liu, Q. Tan, Z. Wang, and B. Zhang, "Sensors, systems and algorithms of 3d reconstruction for smart agriculture and precision farming: A review," *Computers and Electronics in Agriculture*, vol. 224, p. 109229, 2024.

[19] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights into imaging*, vol. 9, pp. 611–629, 2018.

[20] B. Khemani, S. Patil, K. Kotecha, and S. Tanwar, "A review of graph neural networks: concepts, architectures, techniques, challenges, datasets, applications, and future directions," *J Big Data*, vol. 11, no. 1, p. 18, Jan 2024.

[21] C. Shi, X. Wang, and C. Yang, "Advances in graph neural networks," in *Synthesis Lectures on Data Mining and Knowledge Discovery*. Springer, 2022.

[22] J. Xu, Y. Zhang, and D. Miao, "Three-way confusion matrix for classification: A measure driven view," *Information Sciences*, vol. 507, pp. 772–794, 2020.

[23] A. Mosavi, Y. Faghan, P. Ghamisi, P. Duan, S. F. Ardabili, E. Salwana, and S. S. Band, "Comprehensive review of deep reinforcement learning methods and applications in economics," *Mathematics*, vol. 8, no. 10, p. 1640, 2020.

[24] M. Abelha, S. Fernandes, D. Mesquita, F. Seabra, and A. T. Ferreira-Oliveira, "Graduate employability and competence development in higher education-a systematic literature review using prisma," *Sustainability (Switzerland)*, vol. 12, no. 15, p. 5900, 2020.

[25] N. Panic, E. Leoncini, G. de Belvis, W. Ricciardi, and S. Boccia, "Evaluation of the endorsement of the preferred reporting items for systematic reviews and meta-analysis (prisma) statement on the quality of published systematic review and meta-analyses," *PloS one*, vol. 8, no. 12, p. e83138, 2013.

[26] E. Stovold, D. Beecher, R. Foxlee, and A. Noel-Storr, "Study flow diagrams in cochrane systematic review updates: An adapted prisma flow diagram," *Systematic Reviews*, vol. 3, no. 54, pp. 1–5, 2014.

[27] A. Z. Irmansyah, D. Chaerani, and E. Rusyaman, "A systematic review on integer multi-objective adjustable robust counterpart optimization model using benders decomposition," *JTAM (Jurnal Teori dan Aplikasi Matematika)*, vol. 6, no. 3, pp. 678–698, 2022.

[28] D. Chaerani, A. Shuib, T. Perdana, and A. Z. Irmansyah, "Systematic literature review on robust optimization in solving sustainable development goals (sdgs) problems during the covid-19 pandemic," *Sustainability (Switzerland)*, vol. 15, no. 7, p. 5654, 2023.

[29] H. Derviş, "Bibliometric analysis using bibliometrix an r package," *Journal of Scientometric Research*, vol. 8, no. 3, pp. 156–160, 2019.

[30] W. Chen, M. Liu, C. Zhao, X. Li, and Y. Wang, "Mtd-yolo: Multi-task deep convolutional neural network for cherry tomato fruit bunch maturity detection," *Computers and Electronics in Agriculture*, vol. 216, p. 108533, 2024.

[31] C. Tang, D. Chen, X. Wang, X. Ni, Y. Liu, Y. Liu, X. Mao, and S. Wang, "A fine recognition method of strawberry ripeness combining mask r-cnn and region segmentation," *Frontiers in Plant Science*, vol. 14, pp. 1–18, 2023.

[32] R. E. Saragih and A. W. R. Emanuel, "Banana ripeness classification based on deep learning using convolutional neural network," in *2021 3rd East Indonesia Conference on Computer and Information Technology (EIConCIT)*, 2021, pp. 85–89.

[33] Suharjito, M. Asrol, D. N. Utama, F. A. Junior, and Marimin, "Real-time oil palm fruit grading system using smartphone and modified yolov4," *IEEE Access*, vol. 11, pp. 59 758–59 773, 2023.

[34] R. G. Tiwari, H. Maheshwari, A. K. Agarwal, N. K. Trivedi, and V. Gautam, "Federated learning for mango ripeness classification: Evaluating pre-trained deep learning frameworks," in *2023 12th International Conference on System Modeling Advancement in Research Trends (SMART)*, 2023, pp. 23–28.

[35] M. Zhao, Z. You, H. Chen, X. Wang, Y. Ying, and Y. Wang, "Integrated fruit ripeness assessment system based on an artificial olfactory sensor and deep learning," *Foods*, vol. 13, no. 5, p. 793, 2024.

[36] K. Cui, S. Su, J. Cai, and F. Chen, "Walnut ripeness detection based on coupling information and lightweight yolov4," *International Journal of Circuits, Systems and Signal Processing*, vol. 16, pp. 239–247, 2022.

[37] H. Azizi, E. A. Asli-Ardeh, A. Jahanbakhshi, and M. Momeny, "Vision-based strawberry classification using generalized and robust deep networks," *Journal of Agriculture and Food Research*, vol. 15, p. 100931, 2024.

[38] L. Yang, B. Cui, J. Wu, X. Xiao, Y. Luo, Q. Peng, and Y. Zhang, "Automatic detection of banana maturityapplication of image recognition in agricultural production," *Processes*, vol. 12, no. 4, p. 799, 2024.

[39] S. Faza, R. F. Rahmat, M. Husna, R. Anugrahwaty, A. P. Sembiring, and R. P. Ahmad, "Mangrove fruit ripeness classification using deep

convolutional neural network," *Journal of Theoretical and Applied Information Technology*, vol. 101, no. 3, pp. 1095–1105, 2023.
[40] P. Sajitha, A. D. Andrushia, N. Mostafa, A. Y. Shdefat, S. Suni, and N. Anand, "Smart farming application using knowledge embedded-graph convolutional neural network (kegcnn) for banana quality detection," *Journal of Agriculture and Food Research*, vol. 14, p. 100767, 2023.
[41] L. Zhang, Q. Hao, and J. Cao, "Attention-based fine-grained lightweight architecture for fuji apple maturity classification in an open-world orchard environment," *Agriculture*, vol. 13, no. 2, p. 228, 2023.
[42] K. Sumathi and V. Vinod, "Classification of fruits ripeness using cnn with multivariate analysis by sgd," *Neural Network World*, vol. 32, no. 6, p. 19, 2022.
[43] L. Binder, M. Scholz, and R.-D. Kulko, "A comparison of convolutional neural networks and feature-based machine learning methods for the ripeness classification of strawberries," *Bavarian Journal of Applied Sciences*, vol. 2022, pp. 124–137, 2022.
[44] Suharjito, G. N. Elwirehardja, and J. S. Prayoga, "Oil palm fresh fruit bunch ripeness classification on mobile devices using deep learning approaches," *Computers and Electronics in Agriculture*, vol. 188, p. 106359, 2021.
[45] R. Thakur, G. Suryawanshi, H. Patel, and J. Sangoi, "An innovative approach for fruit ripeness classification," in *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 2020, pp. 550–554.
[46] Y. A. Nisa, C. A. Sari, E. H. Rachmawanto, and N. M. Yaacob, "Ambon banana maturity classification based on convolutional neural network (cnn)," *Sinkron: Jurnal dan Penelitian Teknik Informatika*, vol. 8, no. 4, pp. 2568–2578, 2023.
[47] S. Nuanmeesri, "Utilization of multi-channel hybrid deep neural networks for avocado ripeness classification," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 14 862–14 867, 2024.
[48] Z. Xie, Z. Ke, K. Chen, Y. Wang, Y. Tang, and W. Wang, "A lightweight deep learning semantic segmentation model for optical-image-based post-harvest fruit ripeness analysis of sugar apples (annona squamosa)," *Agriculture*, vol. 14, no. 4, p. 591, 2024.
[49] P. Siricharoen, W. Yomsatieankul, and T. Bunsri, "Fruit maturity grading framework for small dataset using single image multi-object sampling and mask r-cnn," *Smart Agricultural Technology*, vol. 3, p. 100130, 2023.
[50] B. Benmouna, G. Garca-Mateos, S. Sabzi, R. Fernandez-Beltran, D. Parras-Burgos, and J. M. Molina-Martnez, "Convolutional neural networks for estimating the ripening state of fuji apples using visible and near-infrared spectroscopy," *Food and Bioprocess Technology*, vol. 15, no. 10, pp. 2226–2236, 2022.
[51] N. Aherwadi, U. Mittal, J. Singla, N. Z. Jhanjhi, A. Yassine, and M. S. Hossain, "Prediction of fruit maturity, quality, and its life using deep learning algorithms," *Electronics*, vol. 11, no. 24, p. 4100, 2022.
[52] A. M. Moraes, L. F. Pugliese, R. F. d. Santos, G. B. Vitor, R. A. d. S. Braga, and F. R. d. Silva, "Effectiveness of yolo architectures in tree detection: Impact of hyperparameter tuning and sgd, adam, and adamw optimizers," *Standards*, vol. 5, no. 1, p. 9, 2025.
[53] K. Han, Y. Wang, J. Guo, Y. Tang, and E. Wu, "Vision gnn: An image is worth graph of nodes," pp. 8291–8303, 2022.