



ISSN: 2723-9535

Available online at www.HighTechJournal.org

HighTech and Innovation Journal

Vol. 4, No. 3, September, 2023



Comparison of CNN Classification Model using Machine Learning with Bayesian Optimizer

Sugiyarto Surono ^{1*}, M. Yahya Firza Afitian ¹, Anggi Setyawan ¹,
Diyah Kresna Eni Arofah ¹, Aris Thobirin ¹

¹ Department of Mathematics, Ahmad Dahlan University, Yogyakarta, Indonesia.

Received 21 June 2023; Revised 17 August 2023; Accepted 28 August 2023; Published 01 September 2023

Abstract

One of the best-known and frequently used areas of Deep Learning in image processing is the Convolutional Neural Network (CNN), which has architectural designs such as Inceptionv3, DenseNet201, Resnet50, and MobileNet used in image classification and pattern recognition. Furthermore, the CNN extracts feature from the image according to the designed architecture and performs classification through the fully connected layer, which executes the Machine Learning (ML) algorithm tasks. Examples of ML that are commonly used include Naive Bayes (NB), k-Nearest Neighbor (k-NN), Support Vector Machine (SVM), and Decision Tree (DT). This research was conducted based on an AI model development background and the need for a system to diagnose COVID-19 quickly and accurately. The aim was to classify the aforementioned CNN models with ML algorithms and compare the models' accuracy before and after Bayesian optimization using CXR lung images with a total of 2000 data. Consequently, the CNN extracted 80% of the training data and 20% for testing, which was assigned to four different ML models for classification with the use of Bayesian optimization to ensure the best accuracy. It was observed that the best model classification was generated by the MobileNetV2-SVM structure with an accuracy of 93%. Therefore, the accuracy obtained using the SVM algorithm is higher than the other three ML algorithms.

Keywords: Classification; Comparison; COVID-19; Convolution Neural Network (CNN); Machine Learning; Bayesian Optimization.

1. Introduction

The COVID-19 virus is a deadly disease caused by Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2). In 2019, the World Health Organization (WHO) declared COVID-19 a new disease that attracted international attention [1]. This virus has been mutating and spreading since August 2021, affecting about 203,295,170 million people worldwide and causing 4,303,515 million deaths [2]. Therefore, an accurate system is needed to quickly and accurately diagnose the virus. This is the reason this research aims to classify COVID-19 by developing a Deep Learning and Machine Learning (ML) model.

One of the best-known and most frequently used Deep Learning approaches in image processing is Convolutional Neural Network (CNN) [3], which has architectural designs such as Inceptionv3, DenseNet201, Resnet50, and MobileNet, which are employed in image classification and pattern recognition [4]. Furthermore, CNN extracts features from the image and performs a classification or regression through a fully connected layer according to the designed ML

* Corresponding author: sugiyarto@math.uad.ac.id

 <http://dx.doi.org/10.28991/HIJ-2023-04-03-05>

➤ This is an open access article under the CC-BY license (<https://creativecommons.org/licenses/by/4.0/>).

© Authors retain all copyrights.

Algorithms. This simply means that a successful classification is achievable using ML algorithms. According to Roihan et al. [5], the best and most frequently utilized ML based on the literature includes Naive Bayes (NB), k-Nearest Neighbor (k-NN), Support Vector Machine (SVM), and Decision Tree (DT).

It is important to note that ML is a supervised and unsupervised algorithm. For example, NB [6] is a supervised ML algorithm that classifies data statistically using the Bayesian method, while k-NN [5] is an unsupervised, as well as uncomplicated, method to understand and use. SVM is a high-precision algorithm widely utilized in the field of bioinformatics with the ability to flexibly handle high-dimensional data [6]. Also, DT [5] is a supervised ML algorithm that classifies big data similar to a tree structure, such as leaves, branches, and nodes. These algorithms are configured to optimally implement ML with the most compatible data and features. Therefore, both supervised and unsupervised ML methods need to be configured before the training process. In this research, hyperparameter optimization (HO) [7] was applied to the ML algorithm in order to achieve the best accuracy. According to Yao et al. [8], the technique improves performance in training steps, prediction accuracy, and ML algorithm quality [8].

In this current research, the CXR lung image data amounted to 2000 and was classified with the ML algorithm, while the Bayesian was used to compare the model's accuracy before and after optimization. First, each CXR chest image feature was extracted using the CNN architectures, namely MobilNetV3, Inception3, ResNet50, and DesNet201. Furthermore, ML methods such as SVM, DT, k-NN, and NB were utilized in classifying the CXR features as COVID-19, Pneumonia, Normal, and Lung Opacity. It was observed that ML methods produced better results; afterward, the hyperparameters of each method were predicted using Bayesian optimization.

2. Literature Review

This section briefly describes several previous studies that performed a DL-based COVID-19 diagnosis using x-ray images. One of them was conducted by Aslan et al. [9], which compared 8 popular CNN architectures, namely Alexnet, Resnet18, Resnet50, Inceptionv3, Densenet201, Inceptionresnetv2, GoogleNet, and MobileNetv2, using 4 ML hyperparameters, such as NB, SVM, DT, and k-NN, with Bayesian optimization. The result showed that Densenet201 produced the highest accuracy of 96.29%. Meanwhile, Yasar & Ceylan [10] tested and compared the classification results of 1,396 lung CT images with the CNN Alexnet and Mobilenetv2 architectures using the k-NN and SVM methods. Das [11] employed U-Net based on the Adaptive Activation Function (AAF-U-Net) as well as SVM, an autoencoder, and NB to replace the fully connected layer in the CNN.

Sethi et al. [12] conducted research on chest X-ray images for the diagnosis of COVID-19 using four different deep CNN architectures, which include Inceptionv3, ResNet50, MobileNet, and Xception. These models are pre-trained with the ImageNet database, thereby reducing the need for large training sets. Among the four models, the one with the highest accuracy results on the MobileNet architecture was 0.986%. In addition, Kundun et al. [13] used the fuzzy integral ensemble method of four deep learning models, namely VGG-11, GoogLeNet, SqueezeNet v1.1, and Wide ResNet-50-2, to classify CT-scan images into the COVID and non-COVID categories. The proposed framework was tested on available data sets, and it achieved 98.93% accuracy and sensitivity.

Ardakani et al. [14] diagnosed COVID and non-COVID-19 from several types of diseases using ten well-known CNN architectures, namely AlexNet, VGG-16, VGG-19, SqueezeNet, GoogleNet, MobileNet-V2, ResNet-18, ResNet-50, ResNet-101, and Xception. Among all architectures, ResNet-101 and Xception produced the best performance. Specifically, ResNet-101 diagnosed COVID-19 from non-COVID-19 cases with sensitivity, specificity, and accuracy of 100%, 99.02%, and 99.51%, while that of Xception was 98.04%, 100%, and 99.02%, respectively. Loey et al. [15] classified chest X-ray images of COVID-19 artifacts using CNN architecture. They further extracted and studied deep features based on Bayesian optimization and tuned CNN hyperparameters according to the objective function. A total of 10,848 datasets utilized were divided into 3 classes, namely COVID-19, normal, and pneumonia, each with 3616 images. From the comparison result of Bayesian optimization with three ablation scenarios, an accuracy of 96% was obtained. Another study by Turkoglu [16] identified and diagnosed COVID-19 disease with the COVIDetectionNet model through a CNN-based AlexNet architecture and performed classification using the SVM method. The total dataset obtained was 6,092 X-ray images, which were classified as normal, COVID-19, and pneumonia, while the result showed an accuracy of 99.18%.

Sameen et al. [17] developed a deep learning-based technique for erosion vulnerability assessment through a one-dimensional convolution network (1D-CNN) and Bayesian optimization to select hyperparameters in South Yangyang Province, South Korea. Random Forest was used to store important factors for further analysis as pre-processing actions; meanwhile, CNN achieved the highest accuracy of 83.11% on the test dataset. Dokeet al. [18] embedded new techniques, such as Bayesian optimization, to efficiently determine the optimal hyperparameter sets. This caused the simple CNN architecture to perform well in the detection of cerebral microbleeds (CMBs). The research employed five CNN layers, namely two convolutions, two pooling, and one fully connected, and the accuracy produced was 98.97%.

3. Materials and Methods

This experimental research was conducted to classify COVID-19 using CNN and ML models, namely SVM, DT, K-NN, and NB. Furthermore, Bayesian optimization was performed to obtain more accurate results, while classification was conducted in the feature extraction stage. This section also contains information regarding the method used, which is expressed in a flowchart as shown in Figure 1.

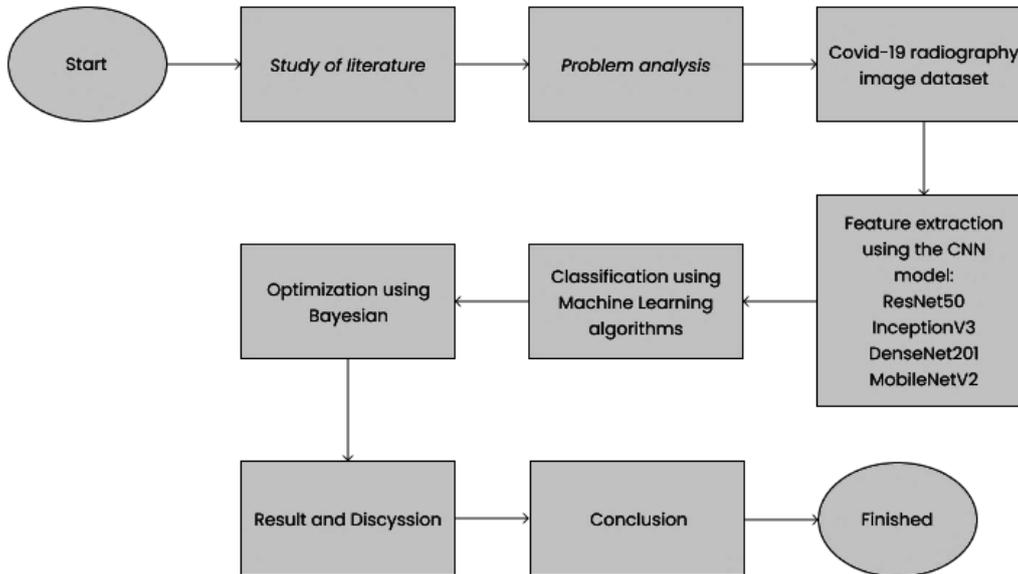


Figure 1. Research flow chart

3.1. Data

The data used was the chest CXR obtained from the kaggle.com website. The dataset was divided into four classes, which include COVID-19, Pneumonia, Lung Opacity, and Normal. Table 1 shows the combination of the classes in the database and the number of images from various sources. Also, the four types of disease classes are shown in Figure 2.

Table 1. Dataset

Class	COVID-19	Normal	Pneumonia	Lung-Opacity
Number of Images CXR	500	500	500	500
<i>Total</i>	<i>2000</i>			

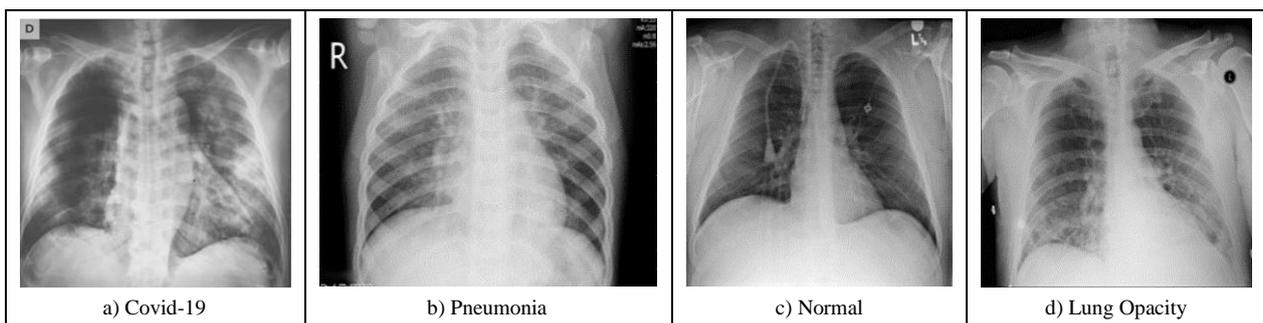


Figure 2. Four classes of disease

3.2. Method

3.2.1. Convolutional Neural Network (CNN)

CNN is a component of deep learning widely applied to image data. In the last decade, it has provided groundbreaking results in areas of pattern recognition, image processing, and speech recognition. It is also capable of extracting features from data by convolution. Meanwhile, the difference between the method and that of traditional feature extraction was that features were not manually extracted [19–21].

3.2.2. Naïve Bayes (NB)

NB is a classification system based on Bayes' theorem, and it assumes that all attributes are completely independent of the output class, known as the conditional independence assumption [22]. According to Sunarya et al. [23], the NB classification algorithm uses the probability theory proposed by British scientist Thomas Bayes, who predicted future probabilities based on previous experience. The main advantage is that it is easy to construct without requiring complex iterative parameter estimation schemes. In addition, the NB classifier is resistant to noise and extraneous properties; hence, it has been successfully applied in many fields [22].

To build a classification for predicting unknown class labels based on Bayes' theorem, let $x = (x^1, x^2, \dots, x^d)$ represents a d-dimensional object with no class label. Also, let $C = \{C^1, C^2, \dots, C^k\}$ be a set of class labels, where $P(C_k)$ is the previous probability of C_k ($k = 1, 2, \dots, K$) concluded before the new evidence, $P(x|C_k)$ denotes the conditional probability of seeing proof x when hypothesis C_k is true. Bayes' theorem was used for the classification as expressed in the following formula [24]:

$$P(C_k|x) = \frac{P(x|C_k)P(C_k)}{\sum_{k'} P(x|C_{k'})P(C_{k'})} \tag{1}$$

where C is Class label set, P is probability, and k is class label set index.

To reduce computations when evaluating $P(X|C_k) P(C_k)$, a naive assumption of class conditional independence was formulated based on the assumption that the attribute values are conditionally independent of each other. Given the sample class label, the mathematical conditional probability is expressed as follows:

$$P(x|C_k) \approx \prod_{j=1}^d P(x|C_k) \tag{2}$$

3.2.3. k-Nearest Neighbours (k-NN)

k-NN is an algorithm used to classify data based on trained datasets obtained from the nearest neighbors, with being the nearest neighbors' number [25]. It performs classification by projecting learning data on a multidimensional space, which is divided into sections representing the learning data criteria. According to Fan et al. [26], each piece of learning data is represented as point c in a multidimensional space. It is important to note that the k-NN is a simple but effective method of categorizing text. However, standard k-NN is a case-based learning technique capable of storing all training data for classification.

3.2.4. Support Vector Machine (SVM)

SVM is a binary classification model used to determine the optimal classification hyperplane that meets the classification requirements [27]. The SVM's goal is to discover the optimal separation hyperplane by maximizing the margin between the separator hyperplane and the data set [28]. From Huang et al. [27], SVM is able to guarantee the hyperplane classification accuracy while maximizing the empty area on both sides of the hyperplane.

$$f(x) = w \cdot x + b = \sum_{k=1}^m w_k \cdot x_k + b = 0 \tag{3}$$

where w and b are weights and biases, respectively, that adjust the position of the hyperplane separator. Meanwhile, the following boundary conditions have to be met by the hyperplane separation:

$$y_i f(x_i) = y_i(w \cdot x_i + b) \geq 1, \quad i = 1, 2, \dots, m \tag{4}$$

The above optimization is converted into a dual quadratic optimization problem using the Lagrangian multiplier α_i , as follows:

$$\text{Maximize } L(\alpha) = \sum_{i=1}^m \alpha_i - \frac{1}{2} \sum_{j=1}^m \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \tag{5}$$

$$\text{s. t } \sum_{i=1}^m \alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad i = 1, 2, \dots, m \tag{6}$$

The dual-problem of the primal problem was obtained by constructing the Lagrange function:

$$\lim_{\alpha} \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) - \sum_{i=1}^N \alpha_i \tag{7}$$

$$\text{s. t } \begin{cases} \sum_{i=1}^N \alpha_i y_i = 0 \\ \alpha_i \geq 0 \quad i = 1, 2, \dots, N \end{cases} \tag{8}$$

where α_i is denoted as Lagrange multiplier.

The advantage of SVM is that it is applicable in non-linear data by modifying the technique using kernel functions [29]. Altan & Karasu [30], the function taking the nonlinear data sequence to a higher dimension is defined as a mapping function and is represented by (Φ) . In this scenario, the regression process was converted to a high-dimensional area through the kernel function.

$$K(x_i, x_j) = \left(\Phi(x_i, x_j)^T \Phi(x_i, x_j) + 1 \right)^p \tag{9}$$

where p is the degree of the polynomial.

3.2.5. Decision Tree (DT)

DT follows a normal tree structure consisting of root, branch, and leaf nodes. Furthermore, attribute testing is performed on each, while the test results of the branch and class labels are found on leaf nodes. It is important to note that a root is the parent of all nodes, and as the name implies, it is the topmost in the tree. DT consists of nodes, which indicate a feature/attribute, with a link/branch representing a decision/rule, and a leaf denoting a result with a categorical or continuous value [31]. The test sample was classified from the root by testing the attribute values at each node and sorting the appropriate branch until it reached the leaf node that provided the classification [32].

Attribute selection steps

- **Entropy**

Entropy is a measure of information theory that detects impurities from the data set. When the attribute identifies different values of c , the entropy S associated with classification c – wise is defined as the equation below [33]:

$$E(S) = \sum_{i=1}^c - P_i \log_2 P_i \tag{10}$$

Where P_i is the ratio S belonging to class i . The entropy is a unit of expected length measured in bits, therefore, the algorithm is expressed as logarithm based 2.

- **Information Gain**

Information Gain selects the attributes used for separating certain nodes. It also prioritizes nominated attributes that have a large number of values by calculating the entropy difference. It is important to note that the Information Gain value is zero when the number of yes or no answers is zero, but when the numbers are equal, the information reaches its maximum. The Information Gain, $Gain(S, A)$ from attribute A , relative to the sample set S , is defined by the following equation [31]:

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{S_v}{S} Entropy(S_v) \tag{11}$$

where $Values(A)$ denotes the set of all potential values for attribute A , while S_v is a subset of S , and attribute A contains a value of v . This measurement is used to group attributes and construct a DT, while each node places the attribute with high Information Gain among those that have not been considered in the path from the root.

- **Gain Ratio**

This is a modification of the information gain that reduces the bias on the high branch attribute. From the equation below, $SplitInfo(D, T)$ is the information due to the separation of T based on the categorical attribute value D [33]:

$$GainRatio(D, T) = \frac{Gain(D, T)}{SplitInfo(D, T)} \tag{12}$$

$$SplitInfo(D, T) = - \sum_{i=1}^K \frac{D_i}{T} \log_2 \frac{D_i}{T} \tag{13}$$

DT is considered one of the most popular classification methods because it is easy to interpret by humans. According to Badriah et al. [34], the basic concept of DT is to convert data into decision trees and rules. It is important to note that DT is one of the powerful methods commonly used in various fields, such as ML, image processing, and pattern recognition. This is consistent with the conclusion in Chen et al. [35] that DT has been implemented in many areas due to its simple analysis and precision on various forms of data [35].

3.2.6. Bayesian Optimization

Bayesian optimization is an approach for optimizing objective functions that take a long time in minutes or hours before being evaluated. Furthermore, it is best suited for optimizing a continuous domain with fewer than 20 dimensions

and tolerates stochastic noise in function evaluation. The approach builds a surrogate for the goal and measures the uncertainty in it using Bayesian ML techniques and Gaussian process regression. Afterward, the acquisition function determined from this surrogate is utilized to decide the place to sample [36].

3.2.7. Feature Extraction

CNN is a mathematical construction consisting of three different layers, namely the convolutional, pooling, and full connecting layers. The first two layers perform feature extraction, while the third maps the extracted features into the final result [36, 37]. The CNN architecture is formed when these layers are put together in different combinations, afterward, the convolution layer is performed by subjecting the input image to a convolution process, and features related to the input image are formed at the output. In this scenario, the pooling layer reduces the parameter number by deriving the output sample from the convolution result. Finally, the full connection layer separates the data into class types after feature extraction. It is important to note that different CNN models have the ability to differentiate layers by combining them with several combination techniques and rules. Table 2 shows the features of the CNN model.

Table 2. CNN model features

Model CNN	Resnet50	Inceptionv3	Densent201	MobileNetv2
Input Size	224 × 224	224 × 224	224 × 224	224 × 224
Feature Layer	fc1000	predictions	fc1000	logits
Number of Extracted Feature	1000	1000	1000	1000

After the input image is fed into the model, this process continues until it reaches the feature layer in the relevant model. The CNN model’s deep features are extracted from a specific layer in each model. It is important to note that the layer used for feature extraction is able to produce different amounts of output depending on the CNN model. For example, the feature extraction layer employed in this current research extracts 2000 features, of which 80% is for training data, while the testing was 20%.

3.3. Bayesian Optimization

Extraction is performed using various features contained in the feature layer, before classifying with the ML algorithm. The method obtains maximum classification, which is very important for determining the parameters affecting the accuracy. This means that the best parameters are determined based on the features used. In this current research, the relevant parameters or hyperparameters were selected using Bayesian optimization in the training step for each ML algorithm. This is because the choices of hyperparameter types differ according to the ML algorithm. Furthermore, the testing process is performed as long as these parameter values are present. In the training phase, the termination criteria are determined by performing a certain number of iterations for each ML algorithm. It was observed that the CNN model and the ML algorithm have different accuracy levels both before and after optimization. The comparison of accuracy before and after Bayesian optimization is shown in Figures 3 to 6.

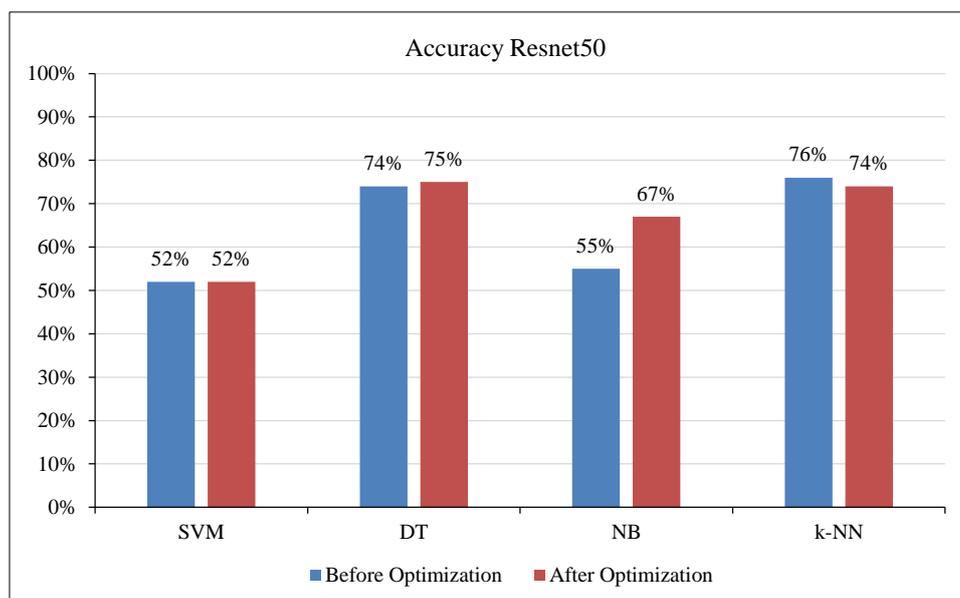


Figure 3. The graph shows the comparison of the accuracy Resnet50 of the optimization results with Bayesian

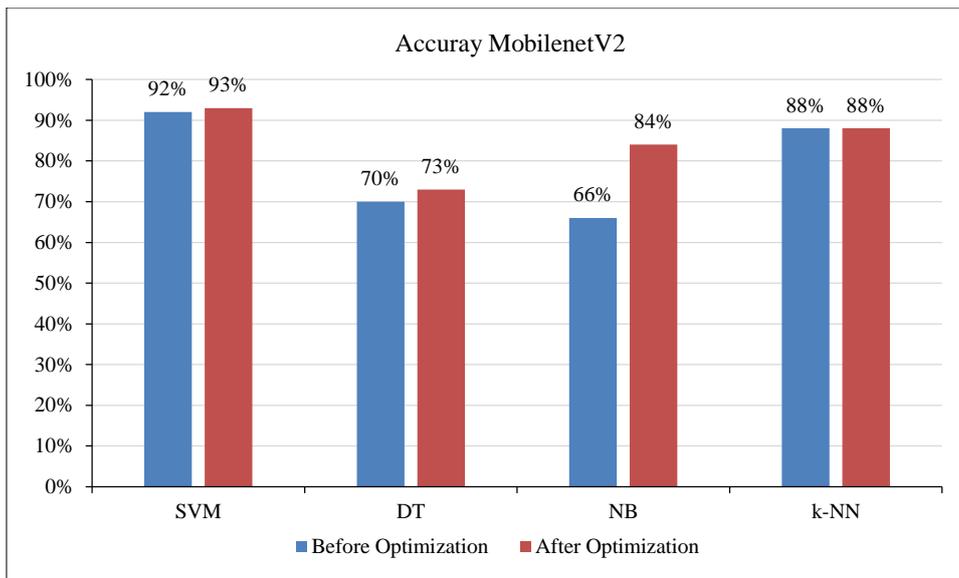


Figure 4. The graph shows the comparison of the accuracy MobilenetV2 of the optimization results with Bayesian

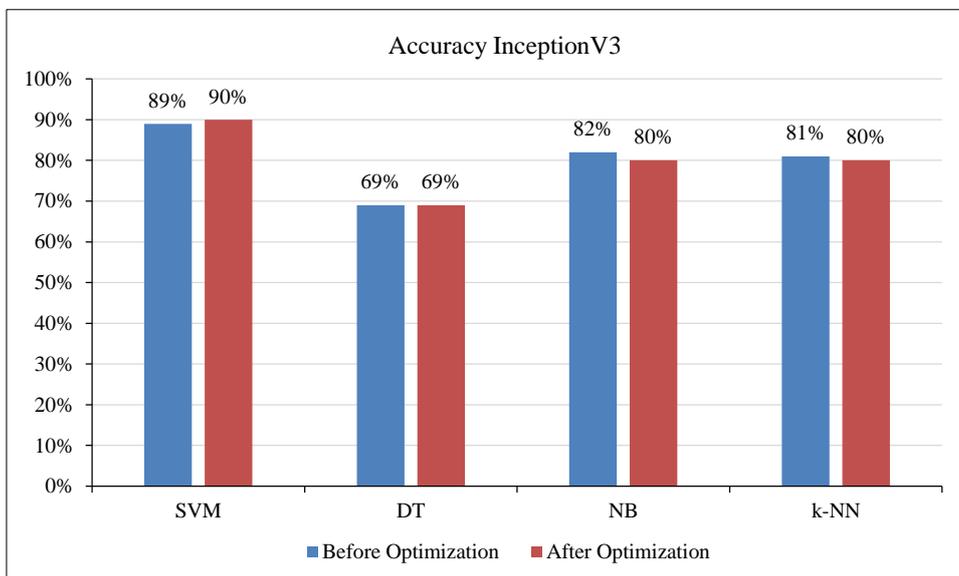


Figure 5. The graph shows the comparison of the accuracy InceptionV3 of the optimization results with Bayesian

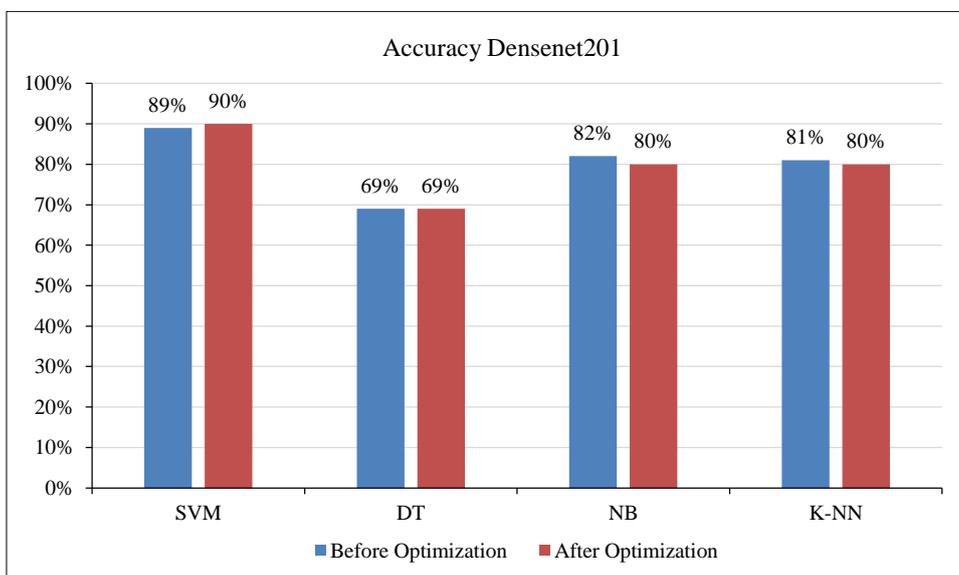


Figure 6. The graph shows the comparison of the accuracy Densenet201 of the optimization results with Bayesian

Based on Figure 3, it was observed that the support vector machine algorithm generates the best optimization accuracy. Table 3 shows the accuracy results as follows.

Table 3. Comparison of accuracy before and after optimization with Bayesian

Classification	Resnet50		MobilenetV2		InceptionV3		Densnet201	
	Before	After	Before	After	Before	After	Before	After
SVM	0.52	0.52	0.92	0.93	0.89	0.90	0.91	0.91
DT	0.74	0.75	0.70	0.73	0.69	0.69	0.73	0.73
NB	0.55	0.67	0.66	0.84	0.82	0.80	0.80	0.83
k-NN	0.76	0.74	0.88	0.88	0.81	0.80	0.85	0.84

According to Table 3, the pre- and post-optimized accuracy with Bayesian on four architectures are as follows. First, on the Resnet50, SVM, DT, NB, and k-NN accuracies before optimization were 52%, 74%, 55%, and 76%, respectively, while after optimization, the values become 52%, 75%, 76%, and 74%. Second, on the MobilenetV2, SVM, DT, NB, and k-NN accuracies before optimization were 92%, 70%, 66%, and 88%, respectively, and after optimization, the values were 93%, 73%, 84%, and 88%. Third, on the InceptionV3, SVM, DT, NB, and k-NN accuracies were 89%, 69%, 82%, and 81%, respectively, while after optimization, the values were 90%, 69%, 80%, and 80%. Finally, on Densnet201, SVM, DT, NB, and k-NN accuracies were 91%, 73%, 80%, and 85%, respectively, and after optimization, the results were 91%, 73%, 83%, and 84%.

4. Results and Discussion

The lung image contains 2000 features, each extracted and classified using four different CNN models and ML algorithms, respectively. In the training step, Bayesian optimizes the parameters affecting each ML algorithm's accuracy. It was observed that the specified hyperparameters do not change during the training and testing steps. Therefore, the confusion matrix is determined based on the classification results as shown in Figure 7.

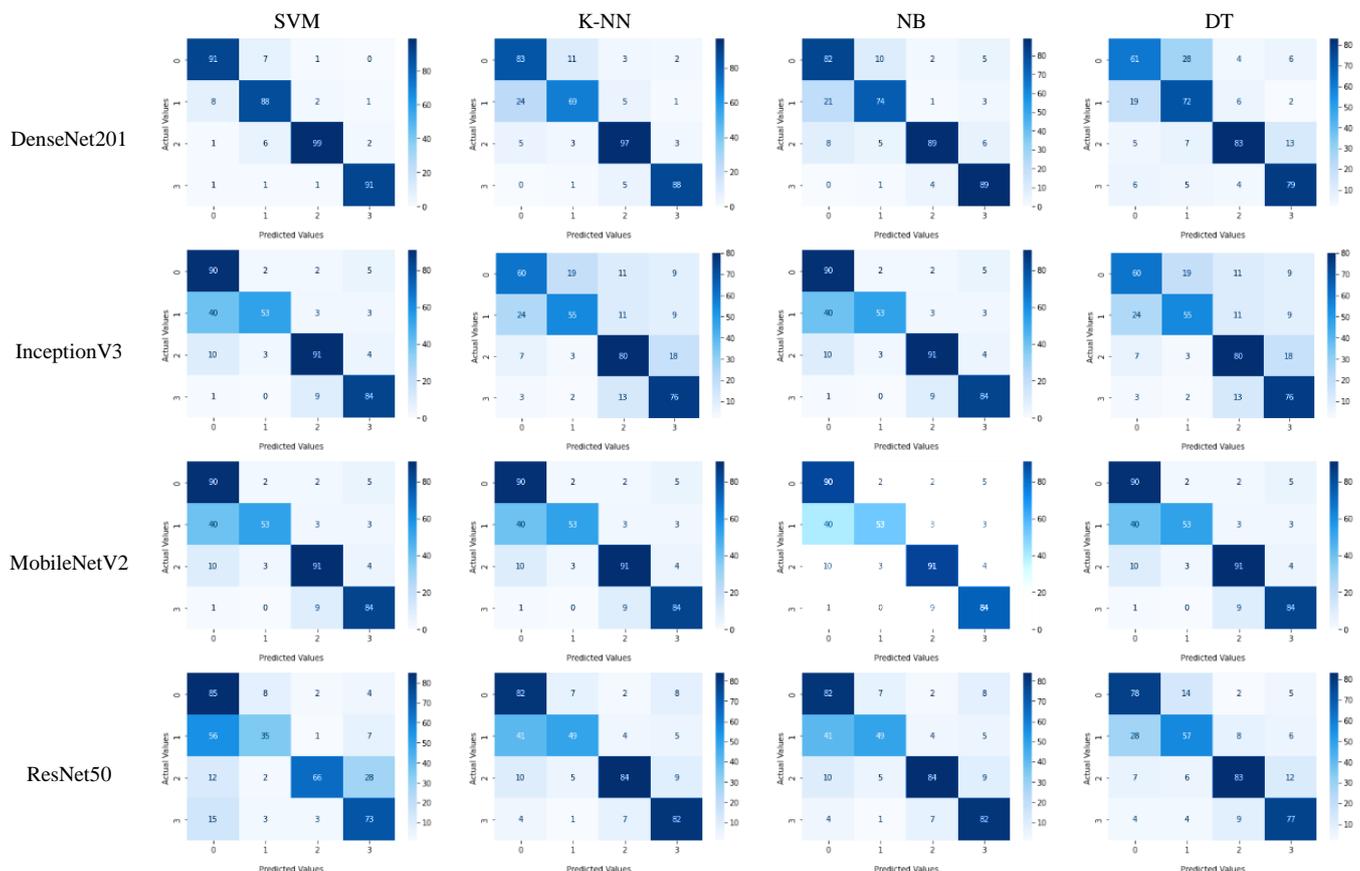


Figure 7. Confusion matrix

Furthermore, different matrices calculated using the confusion matrix are presented to measure the model performance. The following confusion matrix formulas are expressed in Equations 14 to 17 and the results obtained are shown in Table 4.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \times 100 \tag{14}$$

$$\text{Precision} = \frac{TP}{TP+FP} \tag{15}$$

$$F1 \text{ Score} = \frac{2TP}{2TP+FP+FN} \tag{16}$$

$$F1 \text{ Score} = \frac{2TP}{2TP+FP+FN} \tag{17}$$

where TP is True Positive, TN is True Negative, FP is False Positive, and FN is False Negative.

Table 4. The calculation results of the confusion matrix

Model	ML	Accuracy (%)	Precision	Recall	F-1 score
DenseNet201	DT	74%	0.74	0.74	0.74
	NB	83%	0.84	0.83	0.84
	K-NN	84%	0.84	0.84	9.84
	SVM	92%	0.92	0.92	0.92
InceptionV3	DT	68%	0.68	0.68	0.67
	NB	80%	0.82	0.80	0.79
	K-NN	80%	0.82	0.80	0.79
	SVM	90%	0.90	0.90	0.90
MobileNetV2	DT	73%	0.74	0.73	0.73
	NB	84%	0.84	0.84	0.84
	K-NN	88%	0.88	0.88	0.88
	SVM	93%	0.83	0.83	0.83
ResNet50	DT	75%	0.75	0.75	0.75
	NB	67%	0.72	0.67	0.67
	K-NN	74%	0.76	0.74	0.74
	SVM	65%	0.71	0.65	0.64

According to Table 4, the ML algorithm using hyperparameters calculated with Bayesian optimization was successful in classifying each CNN model extracted features. Also, the SVM structure was better in differentiating between classes as the accuracy results are higher than those obtained with other CNN models. When the models are compared, it was observed that the highest accuracy was achieved with MobileNetV2, and among the ML algorithms, the best classification was provided by the SVM with 93% accuracy. The Precision, Recall, and F1-Score values for MobileNetV2-SVM are 0.83 each. Figure 8 shows the confusion matrix obtained with the MobileNetV2 - SVM model.



Figure 8. Confusion matrix model MobileNet-SVM

The results showed that the model proposed in this research for diagnosing COVID-19 was successful and highly accurate.

5. Conclusion

The CNN model extracted 80% of features for training data and 20% for testing. These extracted features are assigned to four different ML algorithms for further classification and were optimized with Bayesian optimization in order to obtain the most accurate result. The pre- and post-optimized Bayesian accuracies are found on four architectures. First, the Resnet50, SVM, DT, NB, and k-NN accuracies before optimization were 52%, 74%, 55%, and 76%, respectively, while after optimization, the values became 52%, 75%, 76%, and 74%. Second, on the MobilenetV2, SVM, DT, NB, and k-NN accuracies before optimization were 92%, 70%, 66%, and 88%, respectively, and after optimization, the values were 93%, 73%, 84%, and 88%. Third, on the InceptionV3, SVM, DT, NB, and k-NN accuracies were 89%, 69%, 82%, and 81%, respectively, while after optimization, the values were 90%, 69%, 80%, and 80%. Finally, on Densnet201, SVM, DT, NB, and k-NN accuracies were 91%, 73%, 80%, and 85%, respectively, and after optimization, the results were 91%, 73%, 83%, and 84%. Based on the confusion matrix calculation, the best classification results using the MobileNetV2-SVM structure give 93% accuracy. The results showed that the SVM was higher than the other three ML algorithms. Therefore, it is concluded that the proposed research for the diagnosis of COVID-19 is very accurate.

The disadvantage of this research is that it provides all scanned images directly as input to the network. Also, the research that uses optimization to improve network prediction performance is still limited. The results indicated that when the lung segmentation and hyperparameter optimization of the ML algorithm are compared with previous research with the same dataset, it provides a significant difference in improving the system performance. This simply means that this current research helps to achieve high classification accuracy through the combination of a state-of-the-art CNN model and an optimized ML algorithm.

Even though this research provided high performance in diagnosing COVID-19, it also has limitations. For example, the proposed model provided high classification accuracy in the COVID-19 Radiographic Database dataset but is likely to show lower accuracy in different datasets. This is because the scanned images in the data set differ from each other due to labels, noise, etc. To solve this problem, AI needs to be trained with scanned images taken at different times and places. In addition to the diversity of the data, the distribution between classes in the data is also important. It was discovered that the imbalance between the numbers of classes negatively affected the training, and the difference in the data augmentation method used to eliminate this imbalance also changed the classification accuracy. Another limitation is that the processing time of the Bayesian optimization was too high, despite improving the success of the ML classification. This simply means that the parameter optimization slows down the diagnosis speed.

6. Declarations

6.1. Author Contributions

Conceptualization, S.S.; methodology, S.S.; software, M.Y.F.A., A.S., and D.K.; validation, S.S.A.T.; formal analysis, S.S.; investigation, A.T.; resources, S.S.; data curation, M.Y.F.A., A.S., and D.K.; writing—original draft preparation, D.K.; writing—review and editing, S.S. and D.K.; visualization, M.Y.F.A.; supervision, S.S.; project administration, A.T.; funding acquisition, S.S. All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

This study was funded by the Ministry of Education and Culture for the fiscal year 2023 with the contract number: 075/E5/PG.02.00.PL/2023,0254.8/LL5-INT/AL.04./2023, 007/PDUPT/LPPM UAD/IV/2023.

6.4. Acknowledgements

The authors are grateful to the Ministry of Education and Culture for funding the PUDPT research for fiscal year 2023 with contract number 075/E5/PG.02.00.PL/2023.

6.5. Institutional Review Board Statement

Not applicable.

6.6. Informed Consent Statement

Informed consent is not applicable to this as the authors used a dataset from Zhang et al. (2020) [24].

6.7. Declaration of Competing Interest

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

7. References

- [1] Harahap, M., Laia, E. M., Sitanggang, L. S., Sinaga, M., Sihombing, D. F., & Husein, A. M. (2022). Detection of Covid-19 Disease in X-Ray Images Using a Convolutional Neural Network (CNN) Approach. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 6(1), 70–77. doi:10.29207/resti.v6i1.3373. (In Indonesian).
- [2] Aslan, N., Ozmen Koca, G., Kobat, M. A., & Dogan, S. (2022). Multi-classification deep CNN model for diagnosing COVID-19 using iterative neighborhood component analysis and iterative ReliefF feature selection techniques with X-ray images. *Chemometrics and Intelligent Laboratory Systems*, 224, 104539. doi:10.1016/j.chemolab.2022.104539.
- [3] Vaishya, R., Javaid, M., Khan, I. H., & Haleem, A. (2020). Artificial Intelligence (AI) applications for COVID-19 pandemic. *Diabetes & Metabolic Syndrome: Clinical Research and Reviews*, 14(4), 337–339. doi:10.1016/j.dsx.2020.04.012.
- [4] Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., & He, Q. (2021). A Comprehensive Survey on Transfer Learning. *Proceedings of the IEEE*, 109(1), 43–76. doi:10.1109/JPROC.2020.3004555.
- [5] Roihan, A., Sunarya, P. A., & Rafika, A. S. (2020). Pemanfaatan Machine Learning Dalam Berbagai Bidang: Review paper. *Indonesian Journal on Computer and Information Technology*, 5(1), 75–82. doi:10.31294/ijcit.v5i1.7951.
- [6] Thakur, S., & Kumar, A. (2021). X-ray and CT-scan-based automated detection and classification of covid-19 using convolutional neural networks (CNN). *Biomedical Signal Processing and Control*, 69. doi:10.1016/j.bspc.2021.102920.
- [7] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 281-305.
- [8] Yao, Y., Cao, J., Ma, Z. (2018). A Cost-Effective Deadline-Constrained Scheduling Strategy for a Hyperparameter Optimization Workflow for Machine Learning Algorithms. *Service-Oriented Computing. ICSOC 2018. Lecture Notes in Computer Science*, 11236. Springer, Cham, Switzerland. doi:10.1007/978-3-030-03596-9_62.
- [9] Aslan, M. F., Sabanci, K., Durdu, A., & Unlarsen, M. F. (2022). COVID-19 diagnosis using state-of-the-art CNN architecture features and Bayesian Optimization. *Computers in Biology and Medicine*, 142(January), 105244, 1-11. doi:10.1016/j.combiomed.2022.105244.
- [10] Yasar, H., & Ceylan, M. (2020). A novel comparative study for detection of Covid-19 on CT lung images using texture analysis, machine learning, and deep learning methods. *Multimedia Tools and Applications*, 80(4), 5423–5447. doi:10.1007/s11042-020-09894-3.
- [11] Das, A. (2021). Adaptive UNet-based Lung Segmentation and Ensemble Learning with CNN-based Deep Features for Automated COVID-19 Diagnosis. *Multimedia Tools and Applications*, 81(4), 5407–5441. doi:10.1007/s11042-021-11787-y.
- [12] Sethi, R., Mehrotra, M., & Sethi, D. (2020). Deep Learning based Diagnosis Recommendation for COVID-19 using Chest X-Rays Images. *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*. doi:10.1109/icirca48905.2020.9183278.
- [13] Kundu, R., Singh, P. K., Mirjalili, S., & Sarkar, R. (2021). COVID-19 detection from lung CT-Scans using a fuzzy integral-based CNN ensemble. *Computers in Biology and Medicine*, 138, 104895. doi:10.1016/j.combiomed.2021.104895.
- [14] Ardakani, A. A., Kanafi, A. R., Acharya, U. R., Khadem, N., & Mohammadi, A. (2020). Application of deep learning technique to manage COVID-19 in routine clinical practice using CT images: Results of 10 convolutional neural networks. *Computers in Biology and Medicine*, 121, 103795. doi:10.1016/j.combiomed.2020.103795.
- [15] Loey, M., El-Sappagh, S., & Mirjalili, S. (2022). Bayesian-based optimized deep learning model to detect COVID-19 patients using chest X-ray image data. *Computers in Biology and Medicine*, 142, 105213. doi:10.1016/j.combiomed.2022.105213.
- [16] Turkoglu, M. (2021). COVIDetectioNet: COVID-19 diagnosis system based on X-ray images using features selected from pre-learned deep features ensemble. *Applied Intelligence*, 51(3), 1213–1226. doi:10.1007/s10489-020-01888-w.
- [17] Sameen, M. I., Pradhan, B., & Lee, S. (2020). Application of convolutional neural networks featuring Bayesian optimization for landslide susceptibility assessment. *Catena*, 186, 104249. doi:10.1016/j.catena.2019.104249.
- [18] Doke, P., Shrivastava, D., Pan, C., Zhou, Q., & Zhang, Y. D. (2020). Using CNN with Bayesian optimization to identify cerebral micro-bleeds. *Machine Vision and Applications*, 31(5), 0–19. doi:10.1007/s00138-020-01087-0.
- [19] Enaizan, O., Saleh, A., Eneizan, B., Almaaitah, M., & Alsakarneh, A. (2022). Understanding and Predicting the Determinants of Consumers' Acceptance and Usage of M-commerce Application: Hybrid SEM and Neural Network Approach. *Emerging Science Journal*, 6(6), 1507-1524. doi:10.28991/ESJ-2022-06-06-018.
- [20] Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2022). A Survey of Convolutional Neural Networks: Analysis, Applications, and Prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 33(12), 6999–7019. doi:10.1109/TNNLS.2021.3084827.
- [21] Akhtar, M. B. (2022). The use of a convolutional neural network in detecting soldering faults from a printed circuit board assembly. *HighTech and Innovation Journal*, 3(1), 1-14. doi:10.28991/HIJ-2022-03-01-01.

- [22] Tien Bui, D., Pradhan, B., Lofman, O., & Revhaug, I. (2012). Landslide susceptibility assessment in Vietnam using support vector machines, decision tree, and naive bayes models. *Mathematical Problems in Engineering*, 2012. doi:10.1155/2012/974638.
- [23] Sunarya, P. O. A., Refianti, R., Mutiara, A. B., & Octaviani, W. (2019). Comparison of accuracy between convolutional neural networks and Naïve Bayes Classifiers in sentiment analysis on Twitter. *International Journal of Advanced Computer Science and Applications*, 10(5), 77–86. doi:10.14569/ijacsa.2019.0100511.
- [24] Zhang, S. (2020). Adaption of Naïve Bayes Classifier in Various Fields. 2020 3rd International Conference on Economic Management and Green Development (ICEMGD 2020), 1-2 August, 2020, Stanford University, Stanford, United States.
- [25] Charbuty, B., & Abdulazeez, A. (2021). Classification Based on Decision Tree Algorithm for Machine Learning. *Journal of Applied Science and Technology Trends*, 2(01), 20–28. doi:10.38094/jastt20165.
- [26] Fan, Z., Xie, J. K., Wang, Z. Y., Liu, P. C., Qu, S. J., & Huo, L. (2021). Image Classification Method Based on Improved KNN Algorithm. *Journal of Physics: Conference Series*, 1930, 012009. doi:10.1088/1742-6596/1930/1/012009.
- [27] Huang, S., Nianguang, C. A. I., Penzuti Pacheco, P., Narandes, S., Wang, Y., & Wayne, X. U. (2018). Applications of support vector machine (SVM) learning in cancer genomics. *Cancer Genomics & Proteomics*, 15(1), 41–51. doi:10.21873/cgp.20063.
- [28] Otchere, D. A., Arbi Ganat, T. O., Gholami, R., & Ridha, S. (2021). Application of supervised machine learning paradigms in the prediction of petroleum reservoir properties: Comparative analysis of ANN and SVM models. *Journal of Petroleum Science and Engineering*, 200, 108182. doi:10.1016/j.petrol.2020.108182.
- [29] Rizal, R. A., Girsang, I. S., & Prasetyo, S. A. (2019). Face Classification Using Support Vector Machine (SVM). *REMIK (Riset Dan E-Jurnal Manajemen Informatika Komputer)*, 3(2), 1. doi:10.33395/remik.v3i2.10080. (In Indonesian).
- [30] Altan, A., & Karasu, S. (2019). The effect of kernel values in support vector machine to forecasting performance of financial time series. *The Journal of Cognitive Systems*, 4(1), 17-21.
- [31] Patel, H. H., & Prajapati, P. (2018). Study and Analysis of Decision Tree Based Classification Algorithms. *International Journal of Computer Sciences and Engineering*, 6(10), 74–78. doi:10.26438/ijcse/v6i10.7478.
- [32] Ghiasi, M. M., & Zendeheboudi, S. (2021). Application of decision tree-based ensemble learning in the classification of breast cancer. *Computers in Biology and Medicine*, 128, 104089. doi:10.1016/j.combiomed.2020.104089.
- [33] Luo, X., Wen, X., Zhou, M. C., Abusorrah, A., & Huang, L. (2022). Decision-Tree-Initialized Dendritic Neuron Model for Fast and Accurate Data Classification. *IEEE Transactions on Neural Networks and Learning Systems*, 33(9), 4173–4183. doi:10.1109/TNNLS.2021.3055991.
- [34] Badriah, S., Nugroho, M. F. E., Sanjaya, N., Rismawati, I., Sari, B. N., & Rozikin, C. (2021). C4 Algorithm Classification. 5 in Determining Recipients of Covid-19 Assistance: (Case Study: Village in Karawang). *Jurnal Informatika Polinema*, 7(3), 23-28. (In Indonesian).
- [35] Chen, W., Yang, G., & Qi, D. (2022). Comprehensive Evaluation of College Students' Physical Health and Sports Mode Recommendation Model Based on Decision Tree Classification Model. *Computational Intelligence and Neuroscience*, 2022, 1–8. doi:10.1155/2022/5504850.
- [36] Otaki, D., Nonaka, H., & Yamada, N. (2022). Thermal design optimization of electronic circuit board layout with transient heating chips by using Bayesian optimization and thermal network model. *International Journal of Heat and Mass Transfer*, 184, 122263. doi:10.1016/j.ijheatmasstransfer.2021.122263.
- [37] Alzubi, Y., Alqawasmeh, H., Al-Kharabsheh, B., & Abed, D. (2022). Applications of Nearest Neighbor Search Algorithm toward Efficient Rubber-Based Solid Waste Management in Concrete. *Civil Engineering Journal*, 8(4), 695-709. doi:10.28991/CEJ-2022-08-04-06.
- [38] Ben Atitallah, S., Driss, M., Boulila, W., & Ben Ghézala, H. (2022). Randomly initialized convolutional neural network for the recognition of COVID-19 using X-ray images. *International Journal of Imaging Systems and Technology*, 32(1), 55–73. doi:10.1002/ima.22654.