Jurnal E-Komtek
Vol. 9, No. 1. (2025) pp. 26-41
https://jurnal.politeknik-kebumen.ac.id/index.php/E-KOMTEK
p-ISSN : 2580-3719 e-ISSN : 2622-3066

Classification of Breast Cancer Magnetic Resonance Imaging (MRI) Using Convolutional Neural Network (CNN) with VGG19 and AlexNet Architecture

Herman Bedi Agtriadi[1], Muhamad Habibi[2], Zakiyah Misfazilah[3]
[1-2]Department of the Engineering Profession, Institut Teknologi PLN, Indonesia, 11750
[3]Department of Master of Computer Science, Institut Teknologi PLN, Indonesia, 11750

hermanbedi@itpln.ac.id

https://doi.org/10.37339/e-komtek.v9i1.2501

Published by Politeknik Piksi Ganesha Indonesia

Artikel Info
Submitted:
17-06-2025
Revised:
25-05-2025
Accepted:
26-05-2025
Online first :
30-06-2025

Abstract

In Indonesia, in 2022 the number of new cases of breast cancer reached 66,271 cases, contributing (30.1%) of the total cancer cases in Indonesia. The number of deaths reached more than 22 thousand people making breast cancer the second most deadly cancer. 70% of breast cancer cases are detected already at an advanced stage, where this case can occur due to delays in medical personnel who have not been able to detect breast cancer manually. This requires technology to help doctors and radiologists to evaluate Magnetic Resonance Imaging (MRI) images automatically. One of the deep learning methods useful for MRI image analysis is Convolutional Neural Network (CNN) using VGG19 and AlexNet architecture which has been proven in the classification process. This study uses data from Kaggle with a total of 1400 data. Through the use of the Convolutional Neural Network method, this study obtained a fairly optimal accuracy on the VGG19 architecture of 99% and on the AlexNet Architecture of 97%.

Keywords*:* Breast Cancer, Convolutional Neural Network, Magnetic Resonance Imaging, VGG19, AlexNet

Abstrak

Di Indonesia, pada tahun 2022 jumlah kasus baru kanker payudara mencapai 66.271 kasus sehingga menyumbang (30,1%) dari total kasus kanker di Indonesia. Jumlah kematian mencapai lebih dari 22 ribu jiwa menempatkan kanker payudara menjadi kanker kedua yang paling mematikan. 70% kasus kanker payudara terdeteksi sudah di tahap stadium lanjut, dimana kasus ini bisa terjadi karena keterlambatan tenaga medis yang belum bisa mendeteksi kanker payudara secara manual. Hal ini dibutuhkan teknologi untuk membantu dokter dan para ahli radiologis untuk mengevaluasi citra Magnetic Resonance Imaging (MRI) secara otomatis. Salah satu metode deep learning yang berguna untuk analisis citra MRI adalah Convolutional Neural Network (CNN) dengan menggunakan arsitektur VGG19 dan AlexNet yang telah terbukti dalam proses klasifikasi. Penelitian ini menggunakan data yang berasal dari Kaggle dengan total 1400 data. Melalui penggunaan metode Convolutional Neural Network penelitian ini memperoleh akurasi yang cukup optimal pada arsitektur VGG19 sebesar 99% dan pada Arsitektur AlexNet sebesar 97%.

Kata-kata kunci*: Kanker Payudara, Convolutional Neural Network, Magnetic Resonance Imaging, VGG19, AlexNet*

1. Introduction

Breast cancer is the most common cancer globally with a malignant category that poses a serious and frightening threat to women. Breast cancer can affect both men and women, but it is most commonly diagnosed in women worldwide. GLOBOCAN (Global Burden of Cancer) statistics in 2022 showed that breast cancer cases in the world reached 58,256,000 cases with 22,692,000 deaths **[1]**. In Indonesia alone in 2022 the number of new cases of breast cancer reached 66,271 thus contributing (30.1%) of the total cancer cases in Indonesia. Of the cases with the number of deaths reaching more than 22 thousand people put breast cancer into the top most deadly cancer.

Magnetic resonance imaging (MRI) is a medical imaging technique that can detect the body using large magnetic fields and radio frequency waves, without surgery and without the use of X-rays, or ultrasound. In the context of breast cancer detection, usually doctors or medical professionals such as radiologists will analyze magnetic resonance images stored in Digital Imagging Communicattion in Medicine (DICOM) format. Sufficient expertise and experience are required in order to provide an accurate diagnosis, so that the treatment carried out can be appropriate[1]. For this reason, a computer-based approach with image processing is used to reduce the risk of errors and increase the accuracy of diagnosis results.

Classification is the process of finding a model or function that distinguishes between concepts or data classes in order to predict the class of an unlabeled object **[2]**. In this research, classification is performed using a method from deep learning: Convolutional Neural Networks (CNN).

Convolutional Neural Network (CNN) is one of the Deep Learning methods that can be used in medical image analysis **[3]**. CNN has the advantage of eliminating the need for an explicit feature extraction process, which is usually required in traditional image processing. CNN has a convolution layer that will automatically detect and extract features from the input image **[4]**. This research uses the VGG19 architecture because its small filter size results in a deeper, more complex network **[5]**.VGG 19 is an architecture consisting of 16 convolutinal layers, 4 max poling layers, 2 fully connected layers, and 1 softmax layer **[6]** and the AlexNet architecture as a comparison in this research offers new solutions to improve the performance of CNNs ranging from ReLU, dropout, and augmentation functions that significantly improve accuracy performance on classification tasks [7]. AlexNet contains 8 layers, consisting of 5 convolutional layers for feature extraction and 3 fully connected layers for classification [8].

This research aims to detect breast cancer in magnetic resonance imaging (MRI) scans using a convolutional neural network (CNN) architecture model. Then, it will determine the accuracy of applying both the VGG19 and AlexNet architecture models to CNN to classify breast cancer in MRIs.

2. Method

Based on MRI image data from Kaggel with a total of 1400 data divided into 2 classes; Benign and Malignant. In the next step, data preprocessing is carried out, namely different image sizes are adjusted to a uniform 224x224 pixels. Next, the dataset is divided into 3 subsets with a ratio of 80% training data, 10% testing data, and 10% validation data.

The structure of a Convolutional Neural Network (CNN) model usually consists of several layers, including the input layer, convolutional layer, ReLU, pooling layer, flatten layer, dropout layer, and fully connected layer [9-13].

2.1 Research Desain

This research uses the Design Scince Research Methodology (DSRM) approach proposed by Peffers et sl, 2006 [14]. DSRM is used in this research stage because it provides a systematic framework for designing, implementing, and assessing Convolutional Neural Network methods, especially in the application of VGG19 and AlexNet.

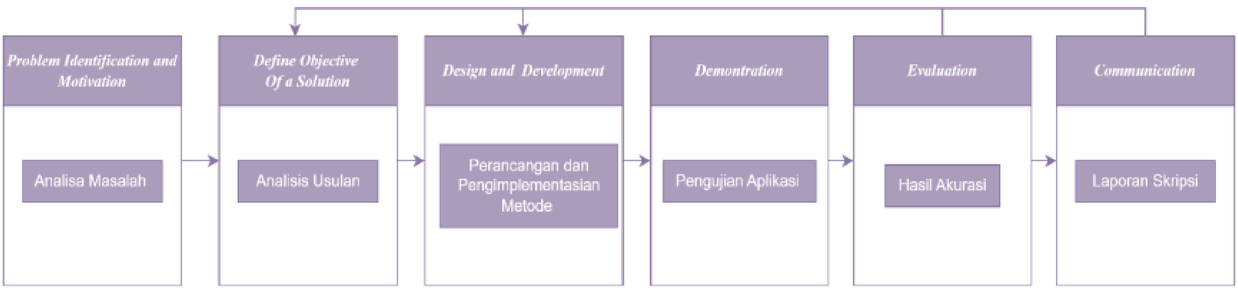


Figure 1. Research Desain

Figure 1 presents the research stages of the DSRM model which are described as follows:

1. Problem Identification and Motivation

The research begins with analyzing the problem, namely looking for a problem from the MRI image of breast cancer. This research is expected to support the process of interpreting MRI images in classifying breast cancer. Through a computerized system that can classify between benign and maglinant categories, this research aims to speed up the diagnosis process and increase its accuracy.

2. Define The Objectives For Solution

After analyzing the problem, the next step is to make a proposal analysis that can help in identifying breast cancer from MRI images. In this research CNN is the suggested technique for image classification.

3. Desain and Development

After carrying out the analysis stage of the proposal, the design and development process is then carried out on the design of the implementation of the method using pyhton language.

4. Demontration

After the design stage and making the implementation of the method are completed, the next step is to measure the accuracy level using the Confusion Matrix method.

5. Evaluation

After carrying out the testing stage, the next evaluation will be carried out. If the test results have met the objectives to be achieved, we can proceed to the next stage.

6. Communication

After obtaining satisfactory evaluation results, the next stage is to make a report containing the results and research documentation for the purposes of the final project report.

2.2 Result of Convolutional Neural Network Model

CNN consists of several layers, including convolutional layer, pooling layer, dropout layer, faltten layer, and fully connected layer. This research starts the first step of system modeling by researching object image data. The image in the Convolutional Neural Network (CNN) stage uses the first input. The data used in this study are images of breast cancer in Magnetic Resonance Imaging (MRI) images which are divided into two classes; benign and malignant.

This research uses 224 x 224 pixel images on both architectures. During the evaluation process, the training data can be used as training data. If the training results are less accurate, the hyperparameters, sample data, and Convolutional Neural Network layers should be changed. If the accuracy is better, it can be used to process test data to determine the type of classification to be performed.

2.3 Training Stages with Convolutional Neural Network Algorithm

The steps used in the Convolutional Neural Network process are as follows:

1. Data Analysis

Image data was collected by using data sources from breast cancer MRI images on the Kaggle website https://www.kaggle.com/datasets/uzairkhan45/breast-cancer-patients-mris. This dataset has 2 types of disease classes (benign and malignant).

2. Training Data

Training data refers to a dataset sourced from kaggle that is used to train and develop models. This research uses a dataset containing 1400 data.

3. Preprocessing

Preprocessing is an important step to transform raw data into a format suitable for effective processing. The preprocessing steps performed will vary depending on the needs of the model **[10]**. In this research, the preprocessing process is divided into two stages, namely:

a. Resize : Resize is the process of changing the size of the image to be used. In this study, researchers resized the default input image with a size of 244 x 244 pixels. This is done so that the size of all images from the dataset used becomes the same so that it facilitates the process to the next stage.

b. Split Data : Split data is the process of dividing data into two or more parts. At this stage, researchers divided the data into 1120 for training data, 140 for testing data, and 140 for validation data with a ratio of 80% training data, 10% testing data, and 10% validation data.

4. Model Training

This research uses the VGG19 and AlexNet models, which of course have different model structures. The following are the structures of each model:

a. Model VGG19

Based on the VGG19 structure, the process starts with the input layer, then continues with the convolution-ReLU and maxpooling processes which are repeated 5 times, followed by the fully connected layer process which is repeated 3 times, and ends with the softmax layer.

1) Convolutional Layer

Convolutional Layer extracts features by calculating the value of each output pixel based on its original value and the values of the surrounding pixels. This calculation involves a kernel (weight matrix) that determines how much influence the surrounding pixels have **[11]**.

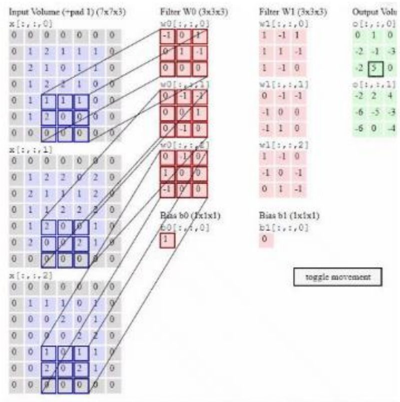


Figure 2. Convolutional Layer

2) Rectified Linear Unit (ReLU) Activation Layer

ReLU (Rectified Linear Unit) activation function is the activation layer in the CNN model, using this function to apply the role meaning f(x)=max(0,x) threshold with zero input image pixel values. The image will be converted to 0 if all *its pixel* values are less than zero [9].

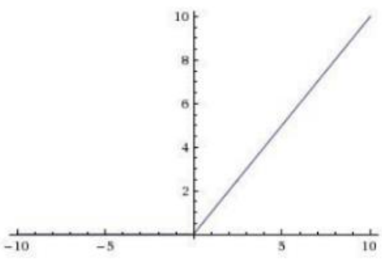


Figure 3. ReLU Activation

3) Pooling Layer

Pooling Layer aims to reduce the size of the *feature map*, or commonly referred to as the subsampling step, which can speed up computation. The pooling layer reduces the size of each feature map stack. The *filters* used are 2x2, applied every 2, and use every *input* slice.

The most commonly used pooling method is Max Pooling. For example, in 2x2 Max Pooling with stride two, the largest pixel value in each 2x2 section will be selected as the filter moves **[12]**.

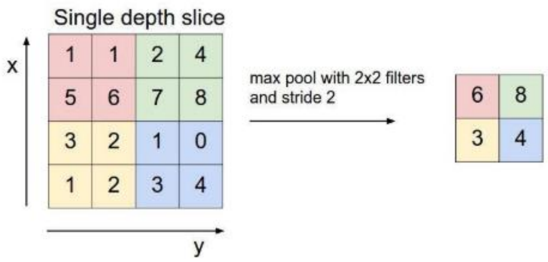


Figure 4. Pooling Layer

4) *Fully Connected layer*

Fully Connected layers are structured like conventional Artificial Neural Networks, consisting of an input layer, one or more hidden layers, and an output layer. Its main feature is dense connectivity, where every neuron in one layer is connected to every neuron in the next layer. Each hidden layer applies an activation function to introduce non-linearity. The main purpose of the fully connected layer is to transform the data to produce accurate image classification, with the final output being the probability for each category if a softmax activation function is used [9].

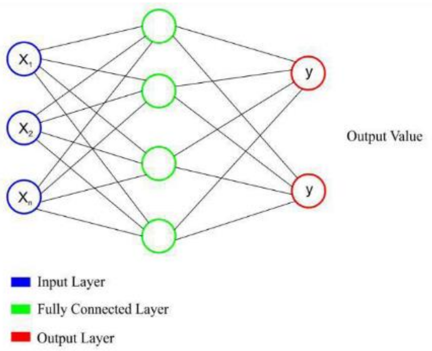


Figure 5. Fully Connected Layer

5) Softmax

softmax function calculates the likelihood of each target class over all possible target classes and helps determine the target class for the input. *Softmax* takes the *E-power* function of the input values and the exponential value of all input values, so the exponential ratio of the input values and the exponential value is the result of the *softmax* function[13]. In softmax the form of the equation that appears is as follows.

$$f_j(Z) = \frac{e^{zj}}{\sum_k e^{zk}} \quad (1)$$

b. Model AlexNet

The structure of AlexNet is almost the same as that of VGG19. However, it has a significant difference in that there are fewer AlexNet layers. The AlexNet structure contains a convolution-ReLU-maxpooling layer that repeats 3 times, followed by fullyconnected-ReLU that repeats 2 times and ends with a softmax layer.

5. Evaluation Model

Model evaluation is carried out to consider if the test results have met the objectives to be achieved or not. Confusion matrix is one of the methods used to determine or detect whether the CNN model has worked well or not. Model testing can be done using *confusion matrix*[2]

The accuracy value displays how accurate the system is to classify the data correctly. The accuracy value is obtained by the formula in equation (2):

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} X\ 100\% \quad (2)$$

Precision displays the ratio of the total number of correctly classified data categories to the total number of correctly classified data categories. The precision value is obtained by the formula in equation (3):

$$Precision = \frac{tp}{tp+tn} \quad (3)$$

Recall features an evaluation metric in machine learning that measures the extent to which a model or system can correctly recognize and predict data that belongs to a positive class. The recall value is obtained by the formula in equation (4):

$$Recall = \frac{tp}{tp+tn} \quad (4)$$

F1-score can calculate *Precision* and *Recall*, with this *F1-score* is very important for *false positives* and *false negatives*. In other words, the *F1-score* indicator affects *false positives* and *false negatives*. The *f1-score* value is obtained by the formula in equation (5):

$$F1\ Score = 2\ x\ \frac{presicion\ x\ recall}{Presicion+recall} \quad (5)$$

3. Result and Discussion

Based on previous literature studies, the current challenge is the increasing number of breast cancer cases each year, which has led to limitations such as a shortage of experts in the field, including doctors and radiologists, and high consultation costs[7]. For this reason, the use of deep learning in the medical field could be one solution. With Deep Learning, medical images can be analyzed to assist doctors and radiologists in the diagnosis process, making it more effective.

3.1. Research Result

This section presents the results of implementing the Convolutional Neural Network method, including the testing process carried out with Python and the evaluation performed using a confusion matrix.

1. Training Process

a. Model VGG19

```
Epoch 1/6
18/18 [==============================] - 27s 662ms/step - loss: 2.0282 - accuracy: 0.5295 - val_loss: 1.2261 - val_accuracy: 0.5000
Epoch 2/6
18/18 [==============================] - 7s 379ms/step - loss: 0.8729 - accuracy: 0.5143 - val_loss: 0.8330 - val_accuracy: 0.5000
Epoch 3/6
18/18 [==============================] - 10s 575ms/step - loss: 0.8815 - accuracy: 0.5607 - val_loss: 0.6372 - val_accuracy: 0.5714
Epoch 4/6
18/18 [==============================] - 7s 384ms/step - loss: 0.6225 - accuracy: 0.6955 - val_loss: 0.4334 - val_accuracy: 0.7357
Epoch 5/6
18/18 [==============================] - 7s 363ms/step - loss: 0.3134 - accuracy: 0.8848 - val_loss: 0.2554 - val_accuracy: 0.8857
Epoch 6/6
18/18 [==============================] - 6s 348ms/step - loss: 0.0832 - accuracy: 0.9670 - val_loss: 0.0590 - val_accuracy: 0.9857
```

Figure 6. VGG19 Training Process

The training process conducted using 6 epochs resulted in an accuracy rate of 0.9670% and a loss of 0.0832%.

b. Model AlexNet

```
Epoch 26/40, Train Loss: 0.1089, Train Acc: 0.9661, Val Loss: 0.2306, Val Acc: 0.9286
Epoch 27/40, Train Loss: 0.0947, Train Acc: 0.9750, Val Loss: 0.1148, Val Acc: 0.9714
Epoch 28/40, Train Loss: 0.0988, Train Acc: 0.9670, Val Loss: 0.1516, Val Acc: 0.9429
Epoch 29/40, Train Loss: 0.0986, Train Acc: 0.9688, Val Loss: 0.1162, Val Acc: 0.9643
Epoch 30/40, Train Loss: 0.0833, Train Acc: 0.9741, Val Loss: 0.1011, Val Acc: 0.9714
Epoch 31/40, Train Loss: 0.0913, Train Acc: 0.9714, Val Loss: 0.1968, Val Acc: 0.9286
Epoch 32/40, Train Loss: 0.0817, Train Acc: 0.9732, Val Loss: 0.0995, Val Acc: 0.9786
Epoch 33/40, Train Loss: 0.0837, Train Acc: 0.9777, Val Loss: 0.1043, Val Acc: 0.9714
Epoch 34/40, Train Loss: 0.0647, Train Acc: 0.9830, Val Loss: 0.1368, Val Acc: 0.9571
Epoch 35/40, Train Loss: 0.0697, Train Acc: 0.9830, Val Loss: 0.1040, Val Acc: 0.9786
Epoch 36/40, Train Loss: 0.0540, Train Acc: 0.9875, Val Loss: 0.1054, Val Acc: 0.9786
Epoch 37/40, Train Loss: 0.0537, Train Acc: 0.9857, Val Loss: 0.0901, Val Acc: 0.9786
Epoch 38/40, Train Loss: 0.0632, Train Acc: 0.9830, Val Loss: 0.0928, Val Acc: 0.9786
Epoch 39/40, Train Loss: 0.0463, Train Acc: 0.9875, Val Loss: 0.1394, Val Acc: 0.9571
Epoch 40/40, Train Loss: 0.0582, Train Acc: 0.9795, Val Loss: 0.0911, Val Acc: 0.9714
Training finished.
```

Figure 7. AlexNet Training Process

The training process conducted using Training Finished with 40 epochs resulted in an accuracy rate of 0.97% and a loss of 0.0582%

2. CNN Training Model Graph

a. Model VGG19

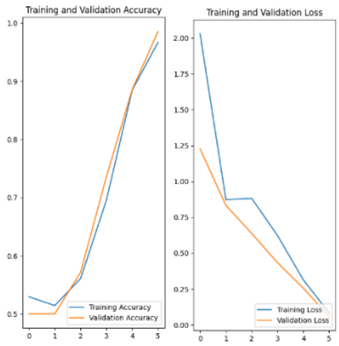


Figure 8. VGG19 Training Graph

Figure 8 shows the VGG19 model, which demonstrates the model's performance during the training process over six epochs, displaying two graphs: accuracy and loss, for both training and validation data. The graph on the left (training and validation accuracy) shows a drastic increase as the epochs progress. This indicates that the model is learning very effectively from the data provided. Meanwhile, the right graph (training and validation loss) decreases sharply from the beginning to the end of training. By the fifth epoch, the loss value approaches zero, indicating that the model is becoming increasingly accurate in making predictions. The consistency between the results on training and validation indicates that the training process is proceeding very well and the model is performing optimally. The values displayed on the X-axis represent the number of training iterations completed during the training process. The values displayed on the Y-axis represent the accuracy rate, with the accuracy value for VGG19 being 99%.

b. Model AlexNet

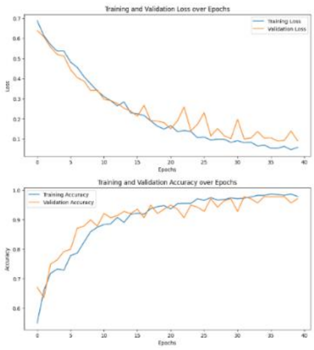


Figure 9. AlexNet Training Graph

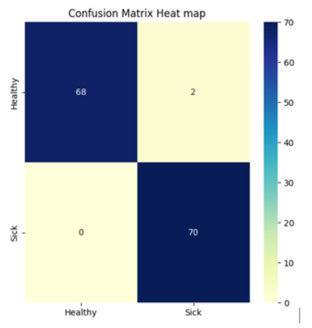
Figure 9 shows the AlexNet model, which demonstrates the model's performance during the training process over 40 epochs, displaying two graphs: accuracy and loss, for both training and validation data. The first graph (training and validation loss) shows that the loss values for both training and validation data generally decrease consistently, indicating that the model is becoming more effective in reducing prediction errors. The second graph (training and validation accuracy) shows that the model's accuracy on both training and validation data experiences a rapid increase in accuracy at the beginning of training and tends to stabilize after around 15 epochs. This accuracy continues to rise until it approaches 1.0, proving that the model has learned effectively and is capable of making highly accurate predictions. The values displayed on the X-axis represent the number of training iterations completed during the training process. The values displayed on the Y-axis represent the accuracy level, with AlexNet achieving an accuracy of 97%.

3. Confusion Matrix

For consideration, the model must be retrained using a confusion matrix. The test results will consist of accuracy, precision, recall, and F1-score values, taking 140 data samples divided into two classes: benign and malignant.

The results of testing the model in each experiment are as follows:

a. Model VGG19

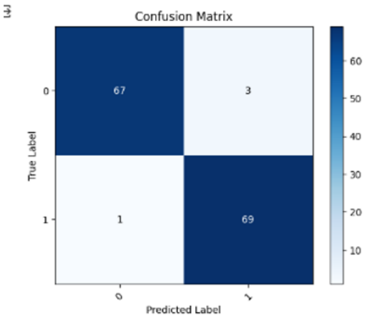


	precision	recall	f1-score	support
0	1.00	0.97	0.99	70
1	0.97	1.00	0.99	70
accuracy			0.99	140
macro avg	0.99	0.99	0.99	140
weighted avg	0.99	0.99	0.99	140

Figure 10. VGG19 Model Testing Result

The model demonstrates excellent performance in classifying both classes (Benign and Malignant) with an overall accuracy of 99%. This model is highly effective in recognizing the Malignant class (Recall 1.00), meaning no Malignant cases were missed. Although there were a few errors (2 cases) where Malignant was classified as Benign, the model has perfect precision (1.00) for the Benign class, indicating that every time the model identifies a case as Benign, the result is always correct.

b. Model AlexNet



Classification Report:

	precision	recall	f1-score	support
0	0.99	0.96	0.97	70
1	0.96	0.99	0.97	70
accuracy			0.97	140
macro avg	0.97	0.97	0.97	140
weighted avg	0.97	0.97	0.97	140

Figure 11. AlexNet Model Testing Result

The model demonstrates excellent performance in classifying both classes (Benign and Malignant) with an overall accuracy of 99%. This model is highly effective in recognizing the Malignant class (Recall 1.00), meaning no Malignant cases were missed. Although there were a few errors (2 cases) where Malignant was classified as Benign, the model has perfect precision (1.00) for the Benign class, indicating that every time the model identifies a case as Benign, the result is always correct.

4. Interface Testing

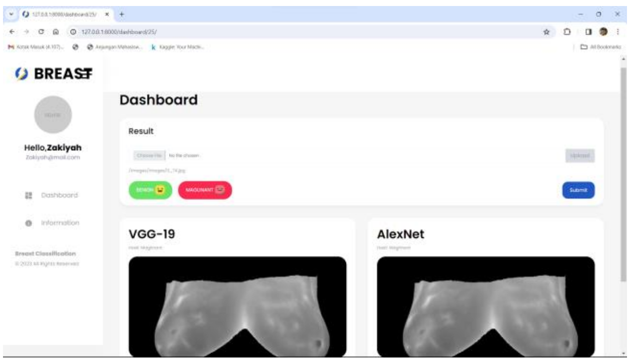


Figure 12. Dashboard Display

This dashboard interface facilitates the testing of CNN implementation for breast cancer detection using MRI imagery. Users are required to select an MRI image in JPG format as the input data. Subsequently, the image will be presented along with its predicted classification as either Benign or Malignant breast cancer, derived from both the VGG19 and AlexNet architectures.

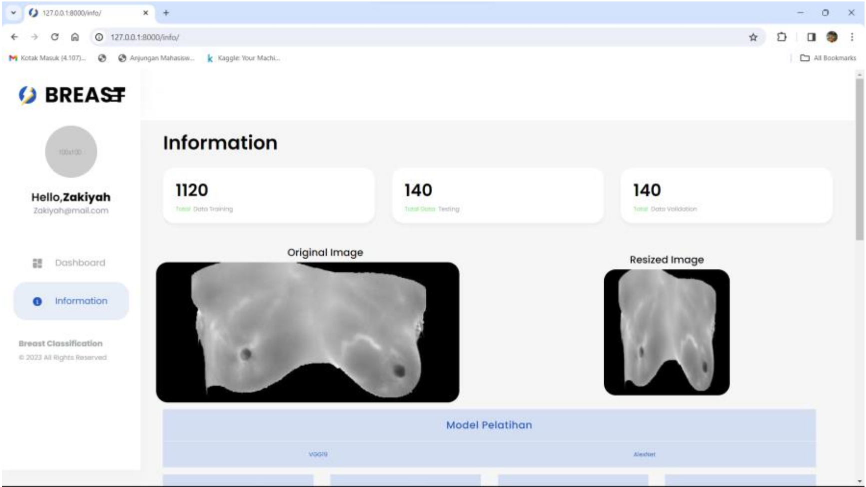


Figure 13. Information Display

This page displays information in the form of data about the classification process, such as data split, resize, training model, classification report, and confusion matrix.

3.2. Discussion

This research focuses on the classification of breast cancer MRI images (Benign and Malignant categories) using the Convolutional Neural Network (CNN) method. The dataset comes from Kaggle, consisting of 1,400 images divided into training (80%), testing (10%), and validation (10%) sets. A comparison was made between two popular CNN architectures: VGG19 and AlexNet.

The analysis shows that VGG19 achieves the highest accuracy of 99%, surpassing AlexNet, which achieved 97%. Furthermore, VGG19 performs very well with a precision of 1, a recall of 0.97, and an F1-score of 0.99. Meanwhile, AlexNet achieved a precision of 0.99, a recall of 0.96, and an F1-score of 0.97. Training duration differs significantly: VGG19 requires only 6 epochs, while AlexNet needs 40 epochs, attributed to the differences in the complexity of their respective architectures. In conclusion, VGG19 demonstrates a clear advantage over AlexNet in this classification task.

In this research, results were obtained from measurements using the CNN method on the VGG19 and AlexNet architectures, as shown in **Table 1**.

Table 1. VGG19 and AlexNet Evaluation

Aktual		**VGG-19**	**AlexNet**
Benign	*Precision*	1.00	0.99
	Recall	0.97	0.96
	F1-Score	0.99	0.97
Malignant	*Precision*	1.00	0.99
	Recall	0.97	0.96
	F1-Score	0.99	0.97
Accuracy		0.99	0.97

4. Conclusions

In the tests conducted, there were a total of 1,400 data points divided into two classes: benign and malignant breast cancer. The dataset was then divided into three subsets with a ratio of 80% training data, 10% validation data, and 10% testing data. After training using two CNN models (VGG19 and AlexNet), the accuracy results were very good, with VGG19 achieving 99% and AlexNet achieving 97%. The conclusions regarding the performance of the two models were influenced by several factors, including the dataset used, the size of the dataset, and the selection of optimal hyperparameters. The hyperparameters used in the creation and training of both models share several similarities, including: the input size used in both architectures is 224x224 pixels; the padding size used in both architectures is the same; the activation function type used after each convolutional layer in both architectures is ReLU (Rectified Linear Unit); the pooling type used in both architectures is Maxpooling; and the activation function used in the output layer of both architectures is softmax. In addition to these similarities, there are several differences in the hyperparameters of the two architectures: the filter size used in the VGG19 architecture is 3x3 for each layer, while the AlexNet architecture uses a filter size of 11x11 in the first layer, 5x5 in the second layer, and 3x3 in subsequent layers. Furthermore, the difference in epochs between the two architectures is quite significant, with VGG19 having 6 epochs and AlexNet reaching 40 epochs.

For future research, it is recommended to use different breast cancer MRI image datasets to evaluate model generalization. This will ensure that the CNN model is not only effective on specific datasets.

References

[1] E. Farijki and B. K. Triwijoyo, "Segmentasi Citra Mri Menggunakan Deteksi Tepi," *J. Matrik*, vol. 16, no. 2, pp. 17–24, 2017.

[2] R. Andre, B. Wahyu, and R. Purbaningtyas, "Klasifikasi Tumor Otak Menggunakan Convolutional Neural Network Dengan Arsitektur Efficientnet-B3," *J. IT*, vol. 11, no. 3, pp. 55–59, 2021, [Online]. Available: https://jurnal.umj.ac.id/index.php/just-it/index.

[3] M. Lestandy, "Deteksi Dini Kanker Payudara Menggunakan Metode Convolution Neural Network (CNN)," *Inspir. J. Teknol. Inf. dan Komun.*, vol. 12, no. 1, p. 65, 2022, doi: 10.35585/inspir.v12i1.2667.

[4] H. Akbar and S. Sandfreni, "Klasifikasi Kanker Serviks Menggunakan Model Convolutional Neural Network Alexnet," *JIKO (Jurnal Inform. dan Komputer)*, vol. 4, no. 1, pp. 44–51, 2021, doi: 10.33387/jiko.v4i1.2606.

[5] Z. Hameed, S. Zahia, B. Garcia-Zapirain, J. J. Aguirre, and A. M. Vanegas, "Breast cancer histopathology image classification using an ensemble of deep learning models," *Sensors (Switzerland)*, vol. 20, no. 16, pp. 1–17, 2020, doi: 10.3390/s20164373.

[6] Darmatasia and M. Syafar, "IMPLEMENTASI CONVOLUTIONAL NEURAL NETWORK UNTUK KLASIFIKASI TANAMAN RIMPANG SECARA VIRTUAL," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 6, no. 2, pp. 666–670, 2023, doi: 10.31539/intecoms.v6i2.6634.

[7] D. Gunawan and H. Setiawan, "Convolutional Neural Network dalam Citra Medis," *KONSTELASI Konvergensi Teknol. dan Sist. Inf.*, vol. 2, no. 2, pp. 376–390, 2022, doi: 10.24002/konstelasi.v2i2.5367.

[8] A. Satyo, B. Karno, W. Hastomo, Y. Efendi, D. Diyah, and R. Irawati, "Konferensi Nasional Ilmu Komputer (KONIK) 2021 Arsitektur Alexnet Convolution Neural Network (CNN) Untuk Mendeteksi Covid-19 Image Chest-Xray," *Konf. Nas. Ilmu Komput.*, vol. 5, no. 1, pp. 482–485, 2021, [Online]. Available: https://prosiding.konik.id/index.php/konik/article/view/105.

[9] F. A. A. Harahap, R. M. Sinaga, K. Arifin, and K. S, "Implementasi Algoritma Convolutional Neural Network Untuk Mendeteksi Penyakit Ginjal Implementation of Convolutional Neural Network for Detecting Kidney Disease," *Teknol. Informasi, Komput. dan Apl.*, vol. 4, no. 2, pp. 212–219, 2022, [Online]. Available: http://jtika.if.unram.ac.id/index.php/JTIKA/.

[10] T. Sulistyorini, E. Sova, N. Sofie, and R. I. Napitupulu, "Penerapan Hyperparameter Convolutional Neural Network (Cnn) Dalam Membangun Model Segmentasi Gambar Menggunakan Arsitektur U-Net Dengan Tensorflow," *J. Ilm. Inform. Komput.*, vol. 28, no. 2, pp. 112–121, 2023, doi: 10.35760/ik.2023.v28i2.6959.

[11] S. Ilahiyah and A. Nilogiri, "Implementasi Deep Learning Pada Identifikasi Jenis Tumbuhan Berdasarkan Citra Daun Menggunakan Convolutional Neural Network _ Ilahiyah _ JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia)," *JUSTINDO(Jurnal Sistem & Teknologi Informasi Indonesia)*, vol. 3, no. 2. pp. 49–56, 2018.

[12] A. Peryanto, A. Yudhana, and R. Umar, "Rancang Bangun Klasifikasi Citra Dengan Teknologi Deep Learning Berbasis Metode Convolutional Neural Network," *Format J. Ilm. Tek. Inform.*, vol. 8, no. 2, p. 138, 2020, doi: 10.22441/format.2019.v8.i2.007.

[13] W. Wijaya Kusuma, R. Rizal Isnanto, A. Fauzi, and P. Korespondensi, "DenseNet121 Menggunakan Kerangka Kerja TensorFlow untuk Deteksi Jenis Hewan," *J. Tek. Komput.*, vol. 1, no. 4, pp. 141–147, 2023, doi: 10.14710/jtk.v1i4.37009.

[14] E. Fernando, S. Surjandy, M. Meyliana, and P. Siagian, "Desain Sistem Pengenalan

Varietas Bibit Tanaman Kelapa Sawit dengan Pendekatan Design Science Research Methodology (DSRM)," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, p. 249, 2020, doi: 10.25126/jtiik.2020721456.