International Journal of Applied Sciences and Smart Technologies

Volume 5, Issue 1, pages 133-142

p-ISSN 2655-8564, e-ISSN 2685-9432

Classification of Lung and Colon Cancer Histopathological Images Using Convolutional Neural Network (CNN) Method on a Pre-Trained Models

Brilly Lutfan Qasthari[1], Erma Susanti[1,*], Muhammad Sholeh[1]

[1]*Faculty of Information Technology and Business, Institut Sains & Teknologi AKPRIND, Yogyakarta, 55222, Indonesia*
[*]*Corresponding Author: erma@akprind.ac.id*

(Received 04-05-2023; Revised 11-05-2023; Accepted 11-05-2023)

Abstract

Cancer is a severe illness that can affect many young and older people. In Indonesia, lung cancer is the leading cause of cancer-related death, whereas colon cancer, with more than 1.8 million cases worldwide in 2018, is the third most common cancer. This study intends to create a model to categorize histological images of lung and colon cancer into five labels to aid medical professionals' categorization job. This study uses a pre-trained model idea known as VGG19 in its CNN (Convolutional Neural Network) technique. The dataset uses 25,000 histological graphic pictures with a ratio of 80% training data and 20% testing data. The classification system for lung and colon cancer contains five categories: lung benign tissue, lung adenocarcinoma, lung squamous cell carcinoma, colon adenocarcinoma, and colon benign tissue. The training result revealed a 99.96% accuracy rate and a 1.5% loss rate. The model can be rated as excellent based on these results.

Keywords: Lung Cancer, Colon Cancer, Convolutional Neural Network, CNN, Pre-Trained

1 Introduction

Cancer is a dangerous condition that can affect both young and older people. Cancer has abnormal characteristics that enable it to target cells or other bodily organs without the affected person knowing it. Estimates of cancer incidence and mortality by sex and for the 18 age groups in 2020 for the 185 countries or regions with a population of more than 150,000 in the same year. When the cells lining the lung airways divide improperly

This work is licensed under a Creative Commons Attribution 4.0 International License

and uncontrolled to generate abnormal tissue, lung cancer results. The most common cancer that causes death is lung cancer [1].

Lung cancer is the leading cause of mortality from cancer in Indonesia [2]. In contrast to colorectal cancer, commonly referred to as colon cancer, this type of cancer develops in the colon, or rectum. The rectum and large intestine are digestive system components of the colon that contribute in the production of energy and the elimination of waste. According to statistics from the American Institute for Cancer Research, colon cancer is the third most prevalent cancer worldwide. There were almost 1.8 million infections in 2018 [3]. In addition to diet, lack of fibre, smoking, and alcohol use, age is the most significant risk factor for colon cancer. Symptoms of colorectal cancer include changes in bowel habits, stomach pain, blood in the stool, anaemia, fatigue, loss of appetite, and weight loss.

Artificial intelligence (AI) technology is used in the medical industry as a decision-support tool for identifying diseases and helps speed up picture analysis. Computer-aided diagnostics can analyze medical photos [3]. The Convolutional Neural Network (CNN) method has been used in several earlier research to demonstrate that cancer may be classified using AI technology, and the resulting model accuracy is good. While compared to manual evaluation by medical experts, AI technology performs computationally more quickly while categorizing lung cancer photos. Modelling the lung cancer categorization system takes two hours of computation [4]. In comparison, a physical examination by medical staff takes 10–14 days to identify lung cancer.

There are several different pre-trained models available on CNN. A few examples of these architectures are Le-Net, Alex-Net, Google-Net, Conv-Net, and Res-Net. In classifying biomedical-based images, the Alex-Net structure is more likely to reach a high accuracy of 90% [5]. In contrast to the ResNet architecture, research utilizing a biomedical-based dataset (Diagnosis of Colonic Adenocarcinoma) was effective in attaining an accuracy of 93% using the ResNet architecture [6].

The pre-trained model is used in this study since it performs well for classification. The average accuracy of the AlexNet and ResNet models is above 90%, based on several prior studies. This work aims to develop a classification system for lung and colon cancer from histological images using various pre-trained models and to assess which model performs best given the histopathological images used.

2 Methods

Three convolutional layers and two fully connected layers will be combined to create the convolutional neural network (CNN) approach for classification in this study. Additionally, it will take advantage of the VGG pre-trained transfer learning architecture. Transfer learning is an approach that makes use of current network infrastructures. There is no need to start from scratch because the CNN architecture utilized for transfer learning has already been learned from prior data. The use of this design will impact the categorization outcomes.

2.1. Convolutional Neural Network

A pooling layer, a few convolutional layers (+ReLU), more convolutional layers (+ReLU), and another pooling layer are common CNN architectures. The image gets smaller and smaller as it moves through the network, but it also usually gets deeper and deeper (i.e., with more feature maps) because of the convolutional layers. The final layer of the stack-for example, a softmax layer that outputs estimated class probabilities - outputs the prediction after adding a standard feedforward neural network made up of a few fully connected layers and ReLUs at the top [7].

2.2. VGGNet

Reusing the lowest layers of a pre-trained model is frequently helpful to develop an image classifier but need more training data. The VGGNet [8] program, created by K. Simonyan and A. Zisserman, was second in the ILSVRC 2014 challenge. It featured a relatively straightforward and traditional design, consisting of 2 or 3 convolutional layers, a pooling layer, 2 or 3 more convolutional layers, a pooling layer, and so on (for a total of just 16 convolutional layers), plus a final dense network with two hidden layers and the output layer. Despite using multiple filters, it only used 3 filters [7].

According to research [9] the CNN architecture produces good results in case study examples of age estimation. The estimating method with a categorization strategy produces satisfactory results. The researchers' challenge with the CNN architecture is to develop the optimum loss function with the most Gaussian distribution. Based on the study's review results, the CNN architecture that provides the best prediction is VGG-16.

2.3. Research Workflow

The research process is shown in Fig. 1's flowchart, which begins with collecting dataset, preprocessing (gathering lung and colon cancer image datasets, scaling images, and dividing datasets), building CNN models with sequential and pre-trained models, training and testing data, storing models, implementing models into the Flask framework, designing the application GUI, and predicting image.

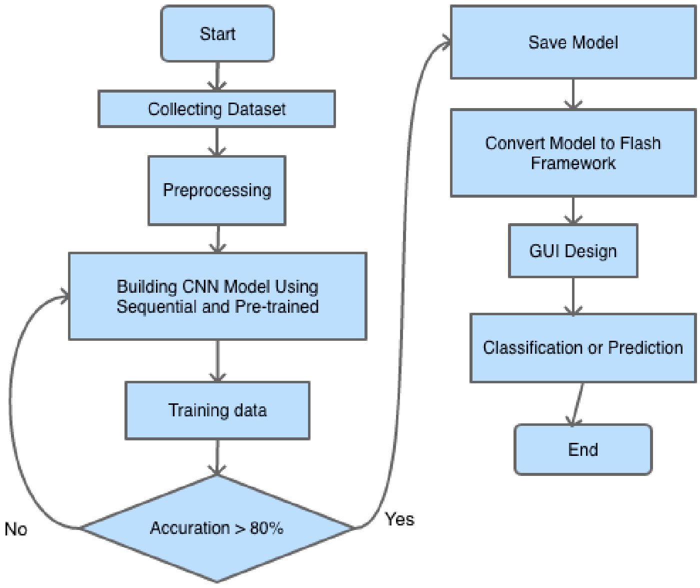

Start
Collecting Dataset
Preprocessing
Building CNN Model Using Sequential and Pre-trained
Training data
Accuration > 80%
No
Yes
Save Model
Convert Model to Flash Framework
GUI Design
Classification or Prediction
End

Figure 1. Workflow Research

3 Results and Discussions

The initial stages of this research involved gathering datasets (Pictured Data), preparing the data, developing a model, testing it, and deploying it. Pre-trained VGG-19-based model is the one being used.

3.1. Collecting Data

The 25,000-image "Lung and Colon Cancer Histopathological Images" dataset from Kaggle served as the source for the image collection. The image collection has five class labels: colon adenocarcinoma, lung squamous cell cancer, benign lung tissue, and colon adenocarcinoma.

3.2. Preprocessing

Data preparation is a step that the user must complete before they edit or add data to a dataset. Because not all incoming data has the same format, the objective is to make understanding easier while reducing confusion during data entry. Preprocessing eliminates the possibility of inaccurate or unnecessary data influencing statistics. 80% of the data are used for training and 20% for testing during the preprocessing stage. The dataset is divided into 20,000 training data for image prediction (training and validation), 4,500 testing data, and 500 dummy data. Afterwards, the information is kept in Google Drive to simplify the image classification process. After that, the picture settings are made, and a data generator is made to produce training and test data.

3.3. Building CNN Architecture Model

Following preprocessing, the next step is the creation of the CNN model. The current study uses pre-trained models. Hence it does not create a model from scratch. The pre-trained principle is to replace the starting layer with the desired layer, often known as fine-tuning. The model can be seen in Table 1.

Table 1. The CNN Model Architecture

No.	*Layer*	*Output shape*	Paramater
1	*Input Layer*	224, 224, 3	0
2	*Layer Vgg19*	0	20.024.384
3	*Global average pooling 2d*	512	0
4	*Flatten*	512	0
5	*Dense*	5120	2.626.560
6	*Dropout*	5120	0
7	*Dense 1*	5	25.605

3.4. Transfer Learning Process

Applying the pre-trained model to carry out the transfer learning process comes after choosing the pre-trained model. The frozen layer on the pre-trained model is where the transfer learning process starts. Information on the frozen layer In the context of CNN, using the frozen layer is how to manage the updated weights. A layer's weight cannot be modified once it has frozen. This method can decrease training data computation time while maintaining accuracy.

3.5. Results of Training and Testing Data

Forward and backward propagation are used during the training phase of the CNN algorithm. Fig. 2 displays the outcomes of training testing and data testing. The model performs well, with a loss on training data of 1.5% and an accuracy of 99.96%.

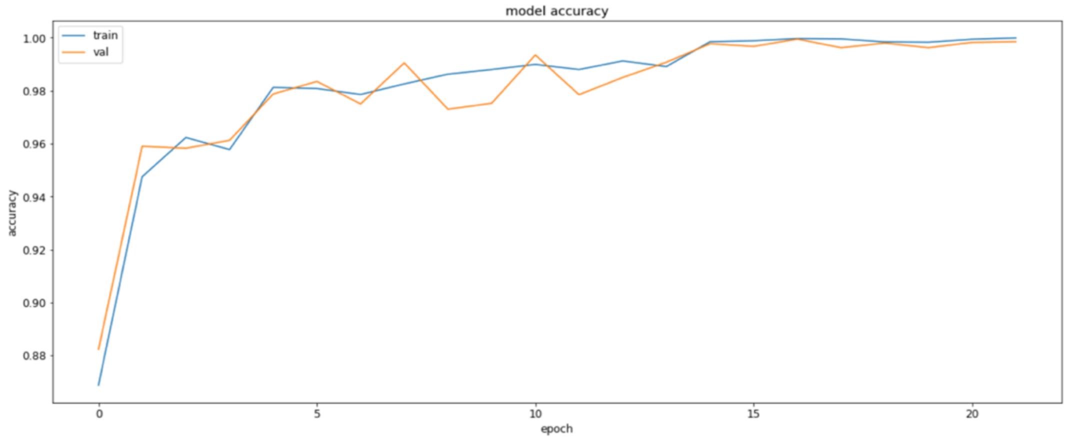


Figure 2. The Result of Accuration Training and Testing Data

After performing data training experiments, the matplotlib package is used in the following epoch iteration to show the training data's outcomes. Fig. 2 provides an excellent model in addition to a good graph. Any model that can run or interpret data without being influenced by noise is good. Because it can describe a trend or set of data with a low error rate, the model will not become overly fitted. This results from the model's high accuracy value and minimal loss. The next step is to show a loss graph from the training data used in the previous step. Four thousand five hundred photos were tested. When used with test data, the model achieves an accuracy performance of 99.82% accuracy and 2% loss, which is identical to that of the training data. The loss outcomes are displayed in Fig. 3. Callbacks have shown to be a very efficient way to reduce the amount of time needed for data training. Training data can yield good accuracy on the 17th epoch and concludes on the 22nd with 30 epochs and 1 hour and 40 minutes of acquisition time.

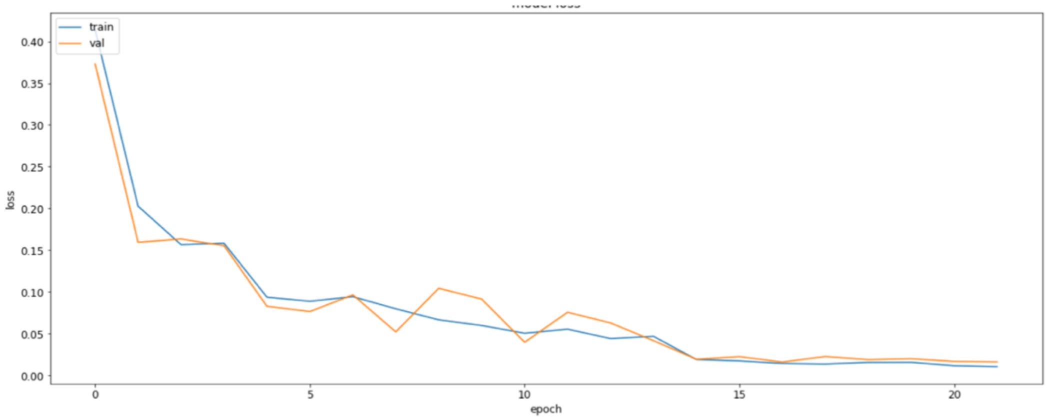
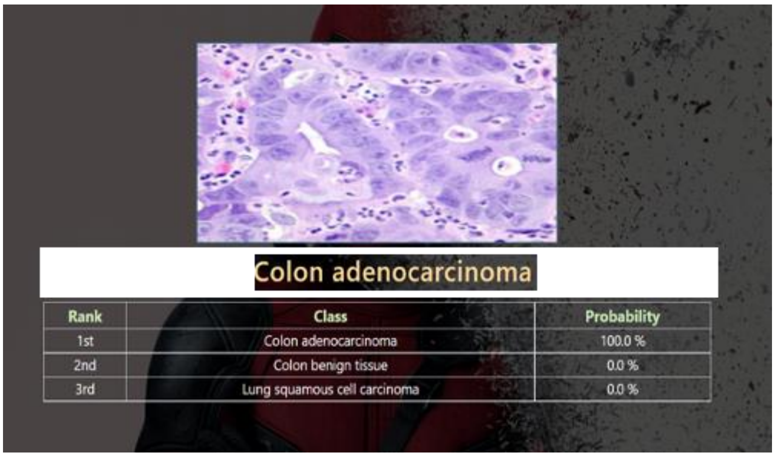


Figure 3. The Result of Loss Training and Testing Data

3.6. Classification or Prediction Result

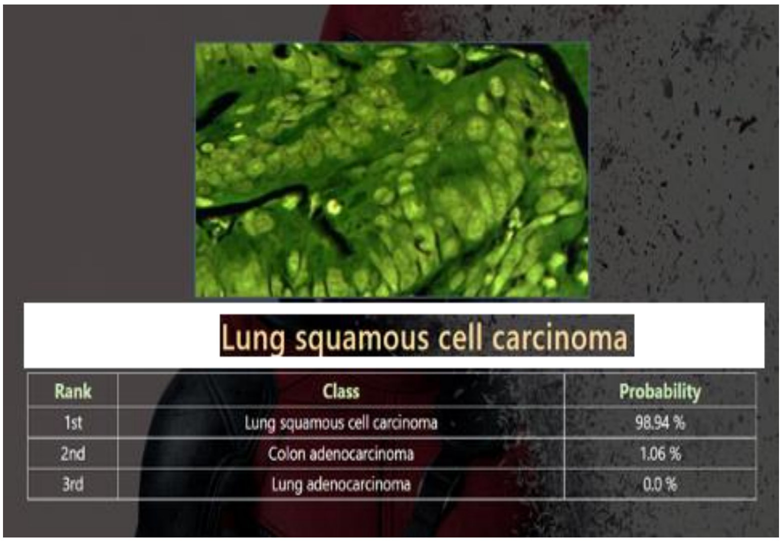
The next step is to test the model by creating image predictions after completing various phases. This prediction test determines whether or not the model can categorize photos. The labels assigned to the prior training data must match the predictions for the images. Fig. 4a illustrates the successful image prediction outcomes using the label for colon adenocarcinoma. Based on the label given, the label prediction in Fig. 4 has a fair chance of coming true. The model correctly predicts the image by the label tested, namely colon adenocarcinoma, with a probability level of 100%. The original image prediction is 768 by 768 in size.



Colon adenocarcinoma

Rank	Class	Probability
1st	Colon adenocarcinoma	100.0 %
2nd	Colon benign tissue	0.0 %
3rd	Lung squamous cell carcinoma	0.0 %

Figure 4. Classification of Colon Adenocarcinoma Label with No Filter



Lung squamous cell carcinoma

Rank	Class	Probability
1st	Lung squamous cell carcinoma	98.94 %
2nd	Colon adenocarcinoma	1.06 %
3rd	Lung adenocarcinoma	0.0 %

Figure 5. Colon Adenocarcinoma Label with Filter

As seen in Fig. 5, the next step is to forecast filtered images using a sample colon adenocarcinoma label with various image forms. The label prediction results could have been better based on the labels given. The filtered image is expected to be 768 × 768 in size before it is resized. The model has not yet forecasted the image per the label examined, namely the colon adenocarcinoma model classifying in the lung squamous cell carcinoma class. Predicting the image that will get the following label is the next stage.

4 Conclusions

The development of the CNN model, which began with the data collection phase and ended with the successful model deployment process, produced good classification results, as shown by the model accuracy results, which reached 99.96% with a 1.5% loss in training data. 99.82% of the data were successfully tested using the evaluate function model, with a 2% loss. Because the model used matches the dataset provided, the outcomes at the feature extraction step of data training utilizing images of colon and lung cancer using the VGG19-based pre-trained model can be considered successful. The use of callbacks can also make it easier to train models that include the checkpoint model feature so that the model can quickly assess the weight that came from the training set of data. The accuracy will increase with the number of epochs used, but there is a significant risk of producing a model with an overfit effect if the number of epochs used is high. The deployment of the model created through the data training procedure on the Flask framework went well. Utilizing the Flask framework, web applications can classify images by the labels provided using models trained on data. The Flask framework can display probability values on the web application interface.

Acknowledgements

The author is grateful for the opportunity to discuss this research with the Study Program and Faculty of Information Technology and Business at IST AKPRIND. We would also like to acknowledge the Python Software, Google Colaboratory, Flask Framework, Keras, Seaborn, and Sklearn packages that were used in this study.

References

[1] J. Ferlay et al., Cancer statistics for the year 2020: An overview, Int. J. Cancer, (2021).

[2] M. G. Sholih et al., Risk factors of lung cancer in Indonesia: A qualitative study, J. Adv. Pharm. Educ. Res., (2019).

[3] D. C. Rini Novitasari, A. Lubab, A. Sawiji, and A. H. Asyhar, Application of feature extraction for breast cancer using one order statistic, glcm, glrlm, and gldm, Adv. Sci. Technol. Eng. Syst., (2019).

[4] R. Apsari, Y. N. Aditya, E. Purwanti, and H. Arof, Development of lung cancer classification system for computed tomography images using artificial neural network, in AIP Conference Proceedings, (2021).

[5] T. Shanthi and R. S. Sabeenian, Modified Alexnet architecture for classification of diabetic retinopathy images, Comput. Electr. Eng., (2019).

[6] S. U. K. Bukhari, A. Syed, S. K. A. Bokhari, S. S. Hussain, S. U. Armaghan, and S. S. H. Shah, The Histological Diagnosis of Colonic Adenocarcinoma by Applying Partial Self Supervised Learning, medRxiv, (2020).

[7] A. Géron, Hands-on Machine Learning. (2017).

[8] Simonyan Karen and Zisserman Andrew, Very deep convolutional networks for large-scale image recognition, in 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, (2015).

[9] P. S. Adi, Development Study of Deep Learning Facial Age Estimation, International Journal of Applied Sciences and Smart Technologies (IJJAST), 1 (1) (2019) 45–50, 2019.