

A convolutional neural network for skin cancer classification

Nur Nafi'iyah¹, Anny Yuniarti²

¹Departement of Informatics Engineering, Lamongan Islamic University, Lamongan, Indonesia

²Departement of Informatics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

Article Info

Article history:

Received Aug 26, 2021

Revised Dec 24, 2021

Accepted Jan 10, 2022

Keywords:

Architecture

Benign

Convolution neural network

Malignant

Skin cancer

ABSTRACT

Skin diseases can be seen clearly by oneself and others. Although this disease is visible on the skin, we fear that this skin disease is harmful. People who experience skin diseases immediately visit a dermatologist to have their complaints and symptoms checked. This skin protects the body, especially from the sun, so it can be lethal if something goes wrong. One example of deadly skin disease is skin cancer or skin tumors. In this research, we classified skin cancer into Benign and Malignant using the convolution neural network (CNN) algorithm. The purpose of this research is to develop the CNN architecture to help identify skin diseases. We used a dataset of 3,297 skin cancer images which are publicly available on the Kaggle website. We propose two CNN architectures that differ in the number of parameters. The first architecture has 6,427,745 parameters, and the second architecture has 2,797,665. The accuracy of the proposed models is 93% and 74% respectively. The first model with the number of parameters 6,427,745 was saved for use in the creation of the website. We created a web-based application with the Django framework for skin disease identification.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Nur Nafi'iyah

Department of Informatics Engineering, Lamongan Islamic University

Veteran 53A St., Lamongan, 62211, Indonesia

Email: mynaff26@gmail.com

1. INTRODUCTION

Deep learning is a neural network model that can help in good computing [1]. Deep learning is also called convolution neural network (CNN). The CNN layers are the convolution layer, activation layer, pooling layer, fully connected layer, and softmax classification [2]. CNN is also used to diagnose skin cancer [2]–[4] and breast cancer classification with an accuracy of 91.3% [5]. CNN is also used to diagnose cervical cancer into seven types of disease and the accuracy is 91.2% to 99.5% [6]. The CNN technique proved significant in dermoscopic melanoma classification with a sensitivity of 95% [7].

Dermatologists need an effective and reliable system for diagnosing skin diseases. Previous researches related to the system used to identify skin diseases are still inefficient because the accuracy value is below 90%. CNN is an accurate and efficient method for identifying skin diseases, to assist dermatologists [8]. With some previous research, we will classify skin cancer using the CNN algorithm. The purpose of this research is to help identify skin diseases early on. Identifying skin diseases early, can help in the treatment and reduce mortality rates. This research previously existed, which was related to the identification of skin diseases using either machine learning or CNN methods. The CNN method produces good accuracy from previous research, so we propose the CNN method to identify skin diseases. This research develops a CNN architecture to identify skin diseases, the best CNN architecture is used as a model in developing a skin disease identification website.

2. RESEARCH METHOD

2.1. Related work

The image of the diseased skin is trained with the Alexnet and VGG-16 architecture so that it can be classified into Benign and Malignant. Melanoma-type skin cancer is a very fatal cancer. To retrieve features from the skin image using principal component analysis (PCA), and wavelet transform [9]. The current algorithm that provides good and reliable accuracy, is CNN. Convolutional neural networks (CNN) are excellent at classifying skin lesions and analyzing images. Diagnosis using a computer with the CNN algorithm can help in the performance of doctors. The framework for the computer-based diagnosis of skin lesions combines the image of the segmented skin lesions and classifies the skin lesions into multiple classes [10]. Classification with convolutional neural networks (CNN) includes accelerated learning (transfer learning), where this process uses an existing network architecture. The transfer learning architecture uses the Inception-v3 pre-trained, resNet-50, Inception-ResNet-v2, and DenseNet-201 [11]. Classification of skin lesions is a process caused by the limitation of the characteristics of the dermoscopic images during the capture or sampling process. There are several types of skin lesions, including cancer such as melanoma, Benign cancer such as nevi, basal cell carcinoma (BCC), and squamous cell carcinoma (SCC) [12]. Convolutional neural networks can be used for the classification of skin lesions in the dermatological field. Image analysis and the process of segmentation and feature extraction of skin lesions must be considered carefully. CNN using a rapid learning architecture (transfer learning) was used to classify skin lesions [13]. CNN is an efficient and accurate method for the analysis of skin disorders. Dermatologists need an effective system to facilitate diagnosis with the ability [14].

Early detection of skin cancer is very important and can prevent death, and several types of skin cancer, carcinoma, and melanoma [15]. A reliable automatic melanoma screening (early detection) system is a system that can perform diagnostics using a computer-based algorithm. The CNN algorithm can be used to screen and detect malignant skin lesions early. The CNN process must require a dataset image along with the skin lesion type as machine learning. The types of skin lesions Balazs research, include melanoma, nevus, and seborrheic [16]. The segmentation of skin lesions is an important process in computer-generated dermoscopic images. There are many segmentation methods for taking the features of skin lesions, one of which is convolutional neural network. The CNN network architecture that is often used for segmentation is (FCN-8s and U-Net) [16]. Computerized convolutional neural networks (CNN) can differentiate melanoma and nevi based on dermoscopic images [17], [18]. 11,444 dermoscopic images were used as the CNN training dataset. The CNN results can be used as a dermatologist's aid in classifying skin lesions on dermoscopic images [18]. Skin cancer is a type of cancer that is often experienced by white people. A good algorithmic approach for the classification or diagnosis of skin lesions is pre-trained CNN [19]. Automatic diagnostic systems for the early detection of skin cancer have had a very good effect [19], [20]. It is proven that the process of treating patients who are detected early can be treated quickly. So that you can make a computerized diagnostic system based on dermoscopic images, you have to do several complete steps. The first step is to segment the skin lesion and remove the dermatoscopic feature. These features are used as a reference for learning convolutional neural networks [20]. Melanoma is a deadly type of skin cancer [21], [22]. So, we need a computer-based system that has a good learning algorithm. Image-based skin cancer detection consists of image repair, segmentation, extraction of interesting features from images, and classification of skin lesions. One good learning algorithm is a convolutional neural network (CNN). CNN can be used to identify malignant tumors on the skin surface with a sensitivity value of 93.3% [22]. This research develops the CNN architecture to identify skin diseases. The best architecture is used as a model to create a skin disease identification website.

2.2. Convolution neural network

This research conducted a classification of Benign and Malignant skin cancer as shown in Figure 1. The dataset image is trained with the CNN algorithm with convolution layer architecture, pooling layer, activation screen, and fully connected. Each screen has a different function, for example, the convolutional screen is used to capture the most interesting image features, as in Figure 1. The pooling screen function of the convolutional feature is taken as the most prominent feature, and the activation screen is to modify or normalize the output. The result of CNN training is a model or weight vector. The model or weight vector is saved to model.h5, then used for testing or testing the classification of skin cancer types. The process of classifying skin cancer on an offline website. How to create an offline website using the framework Django.

Convolution 2D is to multiply the input image with a kernel or filter. The process of multiplying each image pixel will be multiplied by a filter, illustration of multiplication, or a convolution process as in Figure 2(a) and Figure 2(b). The purpose of 2D convolution is to take the maximum features. Figure 2(c) the input image is multiplied by the filter, which changes the size of the input image, initially $n \times m$ to $(n + 2) \times (m + 2)$. The increase in the size of n and m for each edge pixel is given a value of 0. Then each pixel is multiplied by the filter (1):

$$\text{Result_Feature}(i, j) = (\text{image}(i, j) * \text{filter}(i, j)) + (\text{image}(i, j+1) * \text{filter}(i, j+1)) + (\text{image}(i-1, j+1) * \text{filter}(i-1, j+1)) + (\text{image}(i-1, j) * \text{filter}(i-1, j)) + (\text{image}(i-1, j-1) * \text{filter}(i-1, j-1)) + (\text{image}(i, j-1) * \text{filter}(i, j-1)) + (\text{image}(i+1, j-1) * \text{filter}(i+1, j-1)) + (\text{image}(i+1, j) * \text{filter}(i+1, j)) + (\text{image}(i+1, j+1) * \text{filter}(i+1, j+1)).$$

where, i, j are row and column indexes of the image or each image pixel.

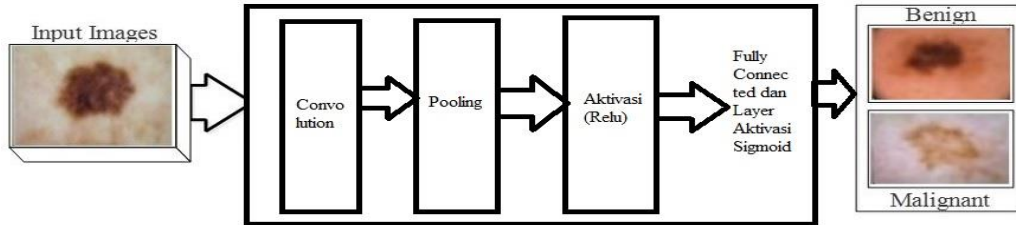


Figure 1. Classification of skin cancer with CNN

Furthermore, the pooling screen is a screen for determining the best feature value, as shown in Figure 3. The image of the feature extraction results is taken for each 2x2 size which is the maximum (Figure 3(a)) or the average (Figure 3(b)). The pooling screen takes the best feature employing the maximum value of each image size or the average value of the image size (Figure 3) and the last screen is a screen for classifying the type of cancer (Benign and Malignant) using the sigmoid function (1).

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

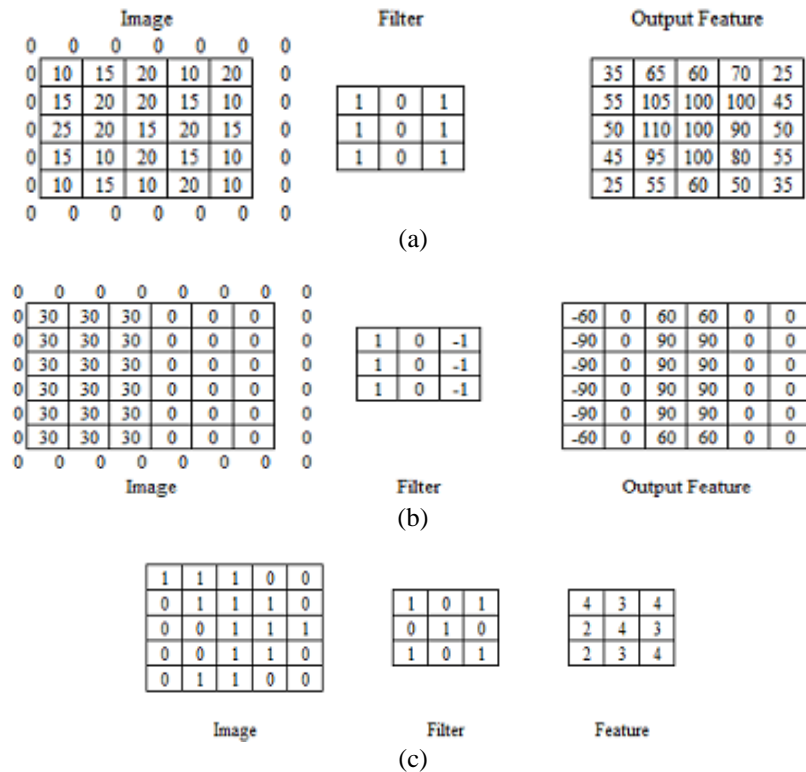


Figure 2. Input image of (a) convolution layer illustration, (b) output feature image size same as input image, and (c) output feature image size smaller

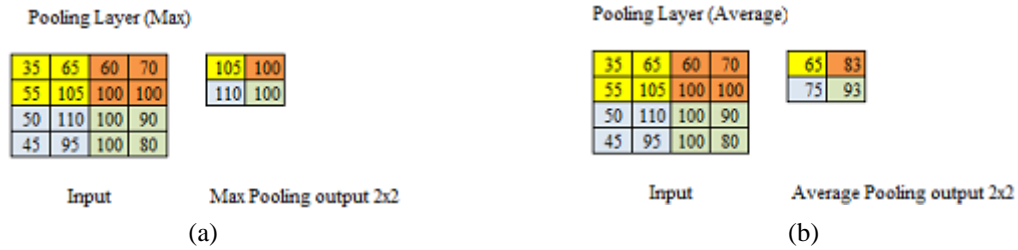


Figure 3. Pooling layer illustration of (a) using max-pooling and (b) average pooling

2.3. Dataset

3297 images used were downloaded from the Kaggle dataset [23], divided as in Table 1. And an example of the dataset used is shown in Figure 4, with an input image size of 224x224 color image types. Image of skin cancer types, namely Benign Figure 4(a), and Malignant Figure 4(b).

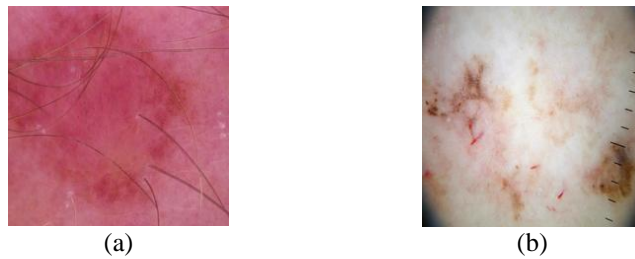


Figure 4. Image of skin cancer types of (a) Benign skin cancer and (b) Malignant skin cancer

Table 1. Skin cancer image dataset

Dataset	Train	Test	Summary
Benign	1440	360	1800
Malignant	1197	300	1497
Summary	2637	660	3297

3. RESULTS AND DISCUSSION

3.1. First CNN architecture

We classified skin cancer into two classes, Benign and Malignant [7]. We classify using two CNN architectural models. The first CNN model architecture has a parameter value of 6,427,745, with architecture like Figure 5. In Figure 5 there are two 2D convolution screens, two pooling screens (using max pooling), and the sigmoid activation function.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 224, 224, 16)	448
max_pooling2d_1 (MaxPooling2)	(None, 112, 112, 16)	0
conv2d_2 (Conv2D)	(None, 112, 112, 32)	4640
max_pooling2d_2 (MaxPooling2)	(None, 56, 56, 32)	0
flatten_1 (Flatten)	(None, 100352)	0
dense_1 (Dense)	(None, 64)	6422592
dense_2 (Dense)	(None, 1)	65
Total params: 6,427,745		

Figure 5. First CNN architecture

The training process for the first CNN model architecture was carried out repeatedly, with as many as 10 epochs and each epoch consisting of 200 iterations. With accuracy values ranging from 85-95%. The training process for each epoch will iterate 200 times, and each iteration will calculate the accuracy value or error value, in order to improve the weight vector. The result of the training process is a model or weight vector (h5), which is then used for classification trials, as in Figure 6. Figure 6 shows how the performance of the training data trials with validation data.

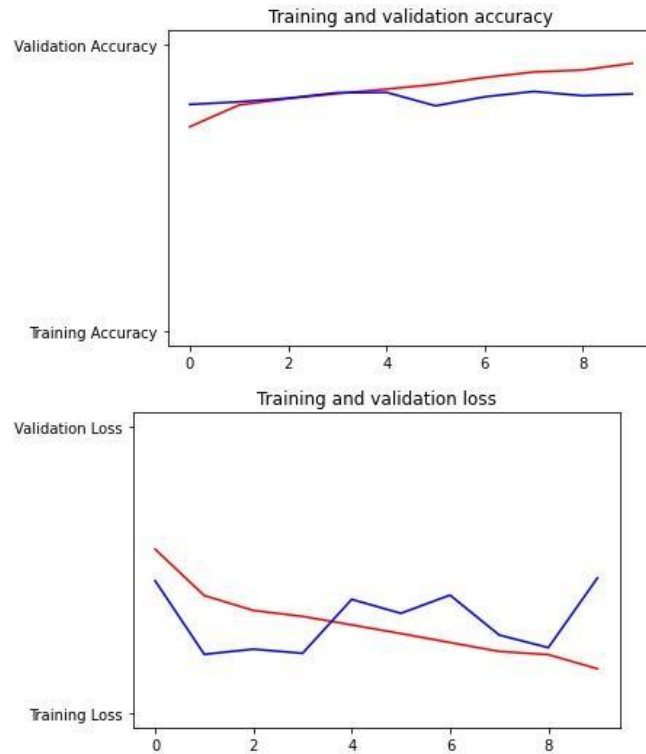


Figure 6. First CNN architectural trial results

3.2. Second CNN architecture

Next, we made the second CNN architecture model with a smaller number of parameters of 2,797,665 as shown in Figure 7. Figure 7 shows a convolution layer three times, three times the pooling screen with max-pooling, and there is a dropout screen. The dropout layer is used to remove some unimportant parameters. From the second CNN architectural model, training was carried out many times. Training is the process of recognizing a pattern or model from an image, which is carried out in as many as 10 epochs and each epoch consists of (100-200 iterations). Each epoch was iterated 200 times and each iteration calculates the accuracy or error, to improve the weight.

The training result is the weight (h5) which is then used for testing or validation. The results of testing or data validation show that the accuracy of the second model is lower because the number of parameters is less as shown in Figure 8. Figure 8 shows that the red lines and blue lines show the results of the accuracy of the validation data and training data.

The testing and training of the proposed CNN algorithm show that a high parameter value will result in high accuracy too. Table 2 shows the results of the testing accuracy of the CNN algorithm that we propose and use the pre-trained. The training used for training and testing CNN were VGG16 [7], Inception-V3, and ResNet50 [9]. The training is a process of transfer learning where the model has been trained with data that has a classification of 1000 classes.

Table 2 shows that with the pre-trained used, the average accuracy result is still low compared to the proposed CNN model. We trained for 10 epochs and each epoch with iterations between 100 and 200 times. The input image that we enter varies in size, but the same type of image is in color. The highest number of parameters is 122,223,521 with the Inception-V3 pre-trained [9] and the size of 224×224 , but the accuracy results are almost the same as the second CNN model we proposed.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 222, 222, 32)	896
activation_1 (Activation)	(None, 222, 222, 32)	0
max_pooling2d_1 (MaxPooling2)	(None, 111, 111, 32)	0
conv2d_2 (Conv2D)	(None, 109, 109, 32)	9248
activation_2 (Activation)	(None, 109, 109, 32)	0
max_pooling2d_2 (MaxPooling2)	(None, 54, 54, 32)	0
conv2d_3 (Conv2D)	(None, 52, 52, 64)	18496
activation_3 (Activation)	(None, 52, 52, 64)	0
max_pooling2d_3 (MaxPooling2)	(None, 26, 26, 64)	0
flatten_1 (Flatten)	(None, 43264)	0
dense_1 (Dense)	(None, 64)	2768960
activation_4 (Activation)	(None, 64)	0
dropout_1 (Dropout)	(None, 64)	0
dense_2 (Dense)	(None, 1)	65
activation_5 (Activation)	(None, 1)	0
Total params: 2,797,665		

Figure 7. Second CNN architecture

Table 2. CNN testing results and transfer learning

CNN type	Number of parameters	Image input	Accuracy
First proposed CNN model	6,427,745	224×224 RGB	93%
Proposed second CNN model	2,797,665	224×224 RGB	73%
VGG16 [7]	23,105,345	150×150 RGB	82%
Inception-V3 [9]	47,512,481	150×150 RGB	77%
Inception-V3 [9]	122,223,521	224×224 RGB	72%
ResNet50 [9]	23,850,242	224×224 RGB	70%

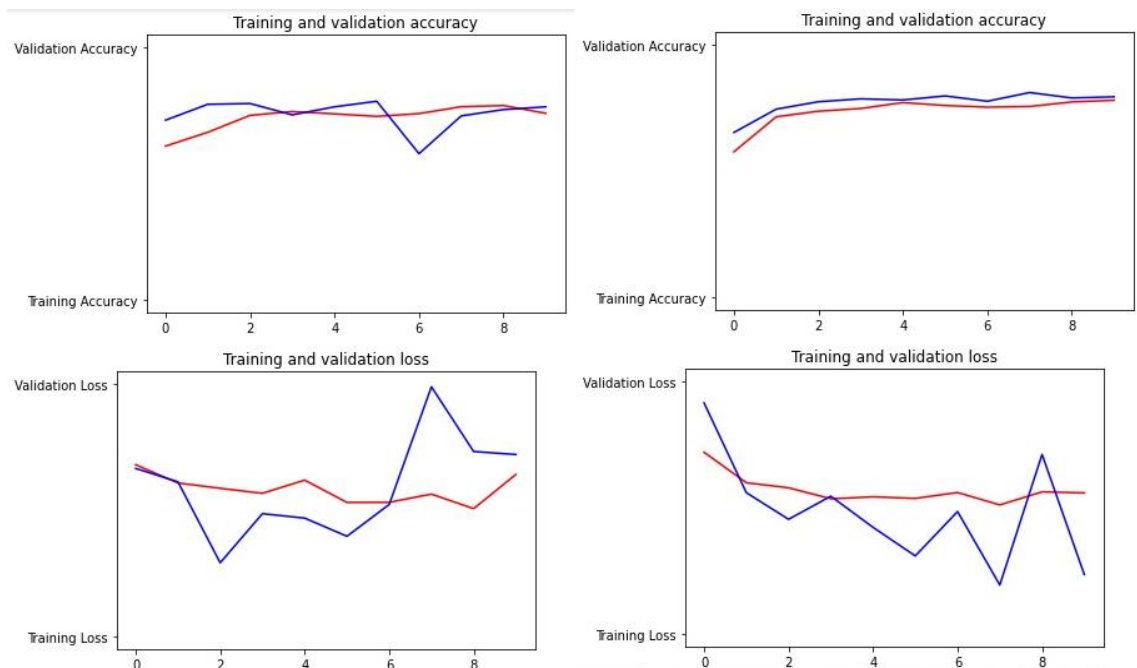


Figure 8. Results of the second CNN architectural trial

3.3. Deploy Django

Furthermore, the results of the model from training are stored in the form (h5) in the form of a weight vector model. Model (h5) is used for web-based classification trials and the web makes it easier for users to test skin cancer classifications (Benign and Malignant) as shown in Figure 9. The following steps are stages to building an offline skin cancer classification website:

- Open a command prompt
- If you have never installed Django, then install Django by writing the command: Conda install Django. Conda is a tool that has many libraries.
- Next, to create a new project, the first step is to write the command: Django-admin startproject name_project
- Move to the newly created name_project folder, with the command: cd name_project
- Create a web application project by writing the command: Django-admin startproject firstApp
- After making commands number 1 to 5, the author of the command or code is carried out in several files, namely: index.html, views.py, urls.py, settings.py

When finished creating a new project and web application or folder according to the steps above, then the following adds several folders and files:

- Adding a media folder, to accommodate the image file of the classification trial results
- Create the models' folder, used to store model.h5 and json files, json files are used to create labels of types or classification classes of skin cancer
- Create a template folder, used to store the index.html file which will appear for the first time when the web is run, as for the architecture of the web folders and files is shown in Figure 10.

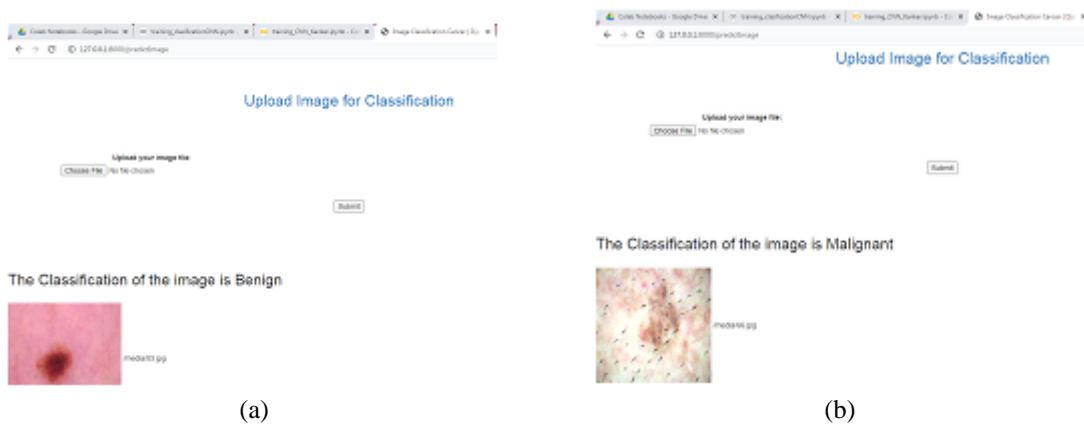


Figure 9. Web views of (a) benign classification test results and (b) Malignant

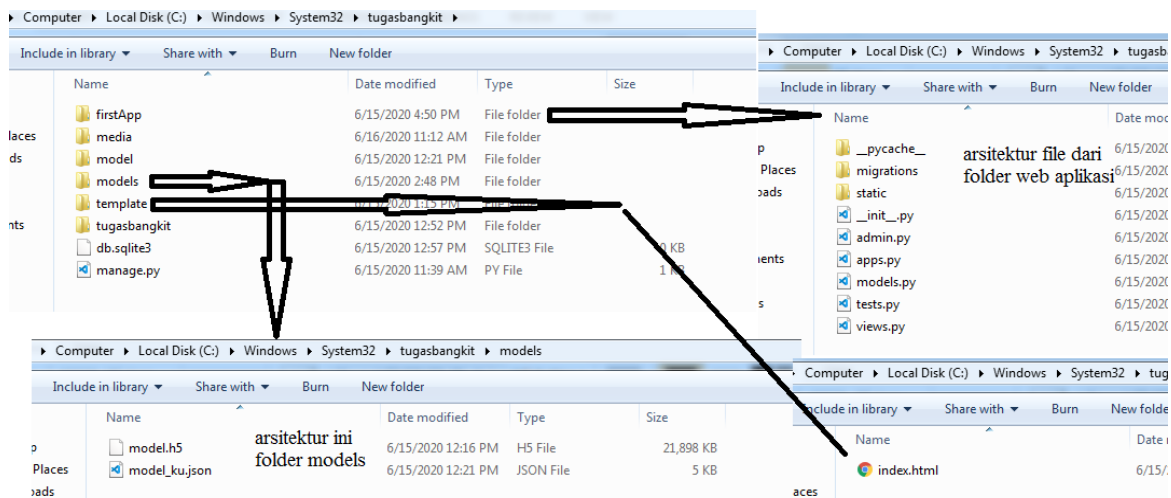


Figure 10. The architecture of a web folder

In figure 10 the index.html file will display the main web page of the application as shown in Figure 9. On the main page (index.html) a menu will appear to select an image file to be tested in a folder or PC drive. Then the image will appear along with the file name as in Figure 9. Click the submit button, the image will be processed and determined the type or class of skin cancer (Benign and Malignant) as depicted in Figure 9. Files that have been submitted will be stored in the media folder as shown in Figure 10. The submit button in Figure 9 will process the classification based on model.h5 in the models' folder. Model.h5 is the CNN training result file. How to determine using the limit value (0.8). This limit value is the value to determine the Benign or Malignant class, if <0.8 then Benign cancer, if not Malignant cancer. This limit value is generated from the testing process many times and seeing the results of the displayed sigmoid value. The results of the sigmoid were then taken as the average value of the two classes. The process for determining the classification of skin cancer is in the views.py file. In order to run the web application, do the following command: i) Open a command prompt; ii) Write the command move to the application folder that was created: `cd nama_folder_project`; and iii) Write the command `python manage.py runserver`. This command will call the file manage.py and will run as long as the web application is restarted as shown in Figure 9.

4. CONCLUSION

The trial results showed that 6,427,745 parameters were able to classify skin cancer with the highest accuracy of 93%. Parameters 2,797,665 were able to classify skin cancer with the highest accuracy of 73%. The number of parameters determines the results of classification accuracy (Benign and Malignant). The number of parameters is determined by the architectural array (CNN layers). Parameter 6,427,745 is the model that has the highest accuracy, then it is stored. The model is used to identify web-based skin diseases with the Django framework. Future research is expected to be able to implement this skin cancer classification with CNN architecture with fewer parameters and high accuracy.

ACKNOWLEDGEMENTS

Thanks to Google and all the Google Bangkit committees.




REFERENCES

- [1] D. de A. Rodrigues, R. F. Ivo, S. C. Satapathy, S. Wang, J. Hemanth, and P. P. R. Filho, "A new approach for classification skin lesion based on transfer learning, deep learning, and IoT system," *Pattern Recognition Letters*, vol. 136, pp. 8–15, Aug. 2020, doi: 10.1016/j.patrec.2020.05.019.
- [2] T. Shanthi, R. S. Sabeenian, and R. Anand, "Automatic diagnosis of skin diseases using convolution neural network," *Microprocessors and Microsystems*, vol. 76, p. 103074, Jul. 2020, doi: 10.1016/j.micpro.2020.103074.
- [3] M. A. Ottom, "Convolutional neural network for diagnosing skin cancer," *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 7, pp. 333–338, 2019, doi: 10.14569/ijacsa.2019.0100746.
- [4] L. Zhang, H. J. Gao, J. Zhang, and B. Badami, "Optimization of the convolutional neural networks for automatic detection of skin cancer," *Open Medicine*, vol. 15, no. 1, pp. 27–37, Jan. 2020, doi: 10.1515/med-2020-0006.
- [5] R. Yan *et al.*, "Breast cancer histopathological image classification using a hybrid deep neural network," *Methods*, vol. 173, pp. 52–60, Feb. 2020, doi: 10.1016/j.ymeth.2019.06.014.
- [6] A. Ghoneim, G. Muhammad, and M. S. Hossain, "Cervical cancer classification using convolutional neural networks and extreme learning machines," *Future Generation Computer Systems*, vol. 102, pp. 643–649, Jan. 2020, doi: 10.1016/j.future.2019.09.015.
- [7] T. J. Brinker *et al.*, "Deep neural networks are superior to dermatologists in melanoma image classification," *European Journal of Cancer*, vol. 119, pp. 11–17, Sep. 2019, doi: 10.1016/j.ejca.2019.05.023.
- [8] K. M. C. Mohammed, S. K. S., and P. G., "Defective texture classification using optimized neural network structure," *Pattern Recognition Letters*, vol. 135, pp. 228–236, Jul. 2020, doi: 10.1016/j.patrec.2020.04.017.
- [9] J. Amin *et al.*, "Integrated design of deep features fusion for localization and classification of skin cancer," *Pattern Recognition Letters*, vol. 131, pp. 63–70, Mar. 2020, doi: 10.1016/j.patrec.2019.11.042.
- [10] H. A. Haenssle *et al.*, "Man against machine reloaded: performance of a market-approved convolutional neural network in classifying a broad spectrum of skin lesions in comparison with 96 dermatologists working under less artificial conditions," *Annals of Oncology*, vol. 31, no. 1, pp. 137–143, Jan. 2020, doi: 10.1016/j.annonc.2019.10.013.
- [11] M. A. Al-masni, D. H. Kim, and T. S. Kim, "Multiple skin lesions diagnostics via integrated deep convolutional networks for segmentation and classification," *Computer Methods and Programs in Biomedicine*, vol. 190, p. 105351, Jul. 2020, doi: 10.1016/j.cmpb.2020.105351.
- [12] Y. Guo and A. S. Ashour, "Neutrosophic multiple deep convolutional neural network for skin dermoscopic image classification," in *Neutrosophic Set in Medical Image Analysis*, Amsterdam, Netherlands: Elsevier, 2019, pp. 269–285.
- [13] K. Sies *et al.*, "Past and present of computer-assisted dermoscopic diagnosis: performance of a conventional image analyser versus a convolutional neural network in a prospective data set of 1,981 skin lesions," *European Journal of Cancer*, vol. 135, pp. 39–46, Aug. 2020, doi: 10.1016/j.ejca.2020.04.043.
- [14] M. Hajabdollahi, R. Esfandiarpour, P. Khadivi, S. M. R. Soroushmehr, N. Karimi, and S. Samavi, "Simplification of neural networks for skin lesion image segmentation using color channel pruning," *Computerized Medical Imaging and Graphics*, vol. 82, p. 101729, Jun. 2020, doi: 10.1016/j.compmedimag.2020.101729.
- [15] N. Zhang, Y. X. Cai, Y. Y. Wang, Y. T. Tian, X. L. Wang, and B. Badami, "Skin cancer diagnosis based on optimized convolutional neural network," *Artificial Intelligence in Medicine*, vol. 102, p. 101756, Jan. 2020, doi:




- 10.1016/j.artmed.2019.101756.
- [16] F. Xie, J. Yang, J. Liu, Z. Jiang, Y. Zheng, and Y. Wang, "Skin lesion segmentation using high-resolution convolutional neural network," *Computer Methods and Programs in Biomedicine*, vol. 186, p. 105241, Apr. 2020, doi: 10.1016/j.cmpb.2019.105241.
- [17] A. Hekler *et al.*, "Superior skin cancer classification by the combination of human and artificial intelligence," *European Journal of Cancer*, vol. 120, pp. 114–121, Oct. 2019, doi: 10.1016/j.ejca.2019.07.019.
- [18] R. C. Maron *et al.*, "Systematic outperformance of 112 dermatologists in multiclass skin cancer image classification by convolutional neural networks," *European Journal of Cancer*, vol. 119, pp. 57–65, Sep. 2019, doi: 10.1016/j.ejca.2019.06.013.
- [19] A. Mahbod, G. Schaefer, C. Wang, G. Dorffner, R. Ecker, and I. Ellinger, "Transfer learning using a multi-scale and multi-network ensemble for skin lesion classification," *Computer Methods and Programs in Biomedicine*, vol. 193, p. 105475, Sep. 2020, doi: 10.1016/j.cmpb.2020.105475.
- [20] M. Pezhman Pour and H. Seker, "Transform domain representation-driven convolutional neural networks for skin lesion segmentation," *Expert Systems with Applications*, vol. 144, p. 113129, Apr. 2020, doi: 10.1016/j.eswa.2019.113129.
- [21] B. Harangi, "Skin lesion classification with ensembles of deep convolutional neural networks," *Journal of Biomedical Informatics*, vol. 86, pp. 25–32, Oct. 2018, doi: 10.1016/j.jbi.2018.08.006.
- [22] V. Srividhya, K. Sujatha, R. S. Ponmagal, G. Durgadevi, and L. Madheshwaran, "Vision based detection and categorization of skin lesions using deep learning neural networks," *Procedia Computer Science*, vol. 171, pp. 1726–1735, 2020, doi: 10.1016/j.procs.2020.04.185.
- [23] Kaggle, "Datasets," Kaggle. [Online]. <https://www.kaggle.com/datasets>.

BIOGRAPHIES OF AUTHORS



Nur Nafiyah    received his first degree from Universitas Islam Lamongan, Informatics Engineering, Lamongan, 2005-2009. He has also Master degree from Sekolah Tinggi Teknik Surabaya, Information Technology, Surabaya, 2011-2013. She can be contacted at email: mynaff@unisla.ac.id.



Anny Yuniarti    received her Bachelor of Computer Science from Institut Teknologi Sepuluh Nopember in 2003 and Master of Computer Science from the University of Western Australia in 2008. She is currently interested in 3D reconstruction from 2D images, deep learning, and computer vision. She also has been teaching computer graphics and human computer interaction courses for the past several years. She can be contacted at email: anny@if.its.ac.id.