Recursive Journal of Informatics
Vol. 1, No. 2, September 2023
p-ISSN 2963-5551 https://journal.unnes.ac.id/sju/index.php/rji e-ISSN 2986-6588

Implementation of Convolutional Neural Network Algorithm Using VGG-16 Architecture For Image Classification In Facial Images

Renita Arianti Hapsari[1], Aji Purwinarko[2]
[1,2]Computer Science Department, Faculty of Mathematics and Natural Sciences,
Universitas Negeri Semarang, Indonesia

Abstract: Face Recognition has now become a technology capable of recognizing facial patterns, facial image recognition is also used in various applications, for example in biological data recognition applications, digital image and video search, room security, and other applications.
Purpose: This study aims to find out how the implementation of the CNN method with the VGG-16 architecture affects the classification of gender in facial images and how it affects the results.
Methods/Study design/approach: In this study, we use the CNN method for data processing and build the program and use VGG-16 Architecture to build the model, then the tensorflow library for calling the required features such as when optimizing or for statistical plots and using the Confusion Matrix to determine the level of accuracy obtained. The desired results in this study are accuracy, precision, recall, and Fscore.
Result/Findings: Classifying facial images using CNN with VGG-16 architecture provides an accuracy rate of 94%. From the results of this study it can be concluded that the model with the best accuracy is at epoch 20 compared to epoch 60, epoch 80, and epoch 100 which have previously been tested.
Novelty/Originality/Value: The level of accuracy resulting from the implementation of the CNN method using the VGG-16 Architecture for image classification in facial images is quite good, resulting in an accuracy of 94%. Accuracy results were obtained from tests carried out by comparing several epoch values to produce the best accuracy of 94% using epoch 20.

Keywords: Image Classification, Face Recognition, Convolutional Neural Network, VGG-16
Received March 30, 2023 / **Revised** April 4, 2023 / **Accepted** September 14, 2023

INTRODUCTION

Face Recognition (facial image recognition) is a technology that can recognize a person's facial pattern. Facial patterns can provide information about a person's identity [1], [2]. Facial image recognition is also part of biometrics that has been studied and developed by many experts. This biometric uses a facial recognition algorithm that associates the identification of a person's face with other individuals whose data has been stored in a separate individual database [3]. Currently, facial image recognition is widely used in various applications, for example in biological data recognition applications, among many facial categorization tasks, gender classification is the most biologically significant [4].

The classification itself is the process of grouping the characteristics of the training images that have been recorded along with the characteristics of the test images [5]. One part of image recognition that has developed so far is gender recognition. The similarity between gender recognition and facial recognition is in the feature extraction process, but the classification process is slightly different. The difficulty in the gender recognition process is mainly caused by the complexity of facial conditions such as image position, lighting, and various image words that have large dimensions and reductions, therefore they must go through a compression or extraction process before processing the data using the classification method [6].

Convolutional Neural Network (CNN) is part of Deep Learning and is a method capable of providing significant results in processing image recognition data. CNN is also reliable in dealing with changes in params when executing image classification [7]. The process to improve time efficiency in the CNN classification process can be done by extracting images from the dataset. To be able to optimize CNN performance in classifying, you can use the VGG 16 architecture. VGG-16 is the best architecture compared to architectures in its class [8].

[1]*Corresponding author.
Email addresses: renitahapsari242@students.unnes.ac.id (Hapsari)
DOI: 10.15294/rji.v1i2.68059

VGG-16 itself is a model that includes image processing layers which include convolution, pooling, and many other layers that can be used to extract images. Where after the images have gone through the processing stage, then after that the images will be tested. After carrying out the next test, accuracy measurements will be carried out. To increase accuracy, researchers use data augmentation so that the input data is more diverse, balances the data [9] and has the potential to increase the sensitivity of the program.

METHODS

This study classifies the face recognition dataset using the CNN algorithm and the VGG-16 architecture. Before the classification process begins, it is necessary to go through the pre-processing stages which are used to increase the variety of data variations and to balance the data. After that the dataset is divided into training and testing data. Before carrying out the classification, two experiments were carried out based on the split data and epochs in the training data process to find the best model. This research is divided into four main steps, as described below.

Data Research

The research data that the researcher used was in the form of facial image files in .jpg file format, totaling 27,167 images. The samples used in this study were 9,489 female facial images and 17,678 male facial images which the researchers obtained from Kaggle.com with the keyword face recognition dataset. Later the dataset is used to train and test the model, by first dividing the dataset into two categories of data, namely Training Data and Testing Data. In each directory there are two classes, namely female and male. Then from the amount of data, a comparison of data distribution will be carried out, namely 70% Training Data: 30% Testing Data, 80% Training Data: 20% Testing Data, and 90% Training Data: 10% Testing Data, the distribution can be seen in Table 1. After that, compare the epoch values, namely epoch 20, 60, 80, and 100 to find the model with the best training results, the best results are seen from the resulting accuracy. Researchers used the CNN method with the VGG-16 architecture in this two experiments.

Table 1. Data sharing

	Male		Female		Amount	
Comparison	Training Data	Testing Data	Training Data	Testing Data	Training Data	Testing Data
70:30	20.685	8.885	20.615	8.815	41.300	17.700
80:20	23.643	5.943	23.557	5.857	47.200	11.800
90:10	26.574	2.974	26.526	2.926	53.100	5.900

Pre-Processing

This stage is necessary to prepare data that was previously raw so that it can be processed first, so that later the data quality will be better and the model can optimize data processing.

These steps include:

1. Rescale is a command that is used to change the scale value in the image. In this study, the value of the image scale is (1/255), which means that each value is multiplied (1/255) so that the final result is 1 and 0 in order to simplify the Training Data process.
2. Resize (resize) may look the same as rescale, but resize itself does not change the value on the scale but changes the value of the data size (resolution) in the image. Simply put, rescaling is able to reduce the image without reducing the data size, while resizing is the opposite [10]. In this study, resizing is needed to equalize the size between images or each image, because in the image dataset used the size of each image is different, therefore it still needs to be equated. The size of the image will later be resized to (100, 100), so that the size or dimensions of each image are the same.
3. Data augmentation is a process in processing image data, augmentation itself is a process of modifying images in such a way that the computer detects that the changed image is a different image, but humans are still able to know the changed image is the same image [11]. Augmentation can increase the accuracy of the CNN model being trained because using the augmentation method the model obtains more diverse data so as to increase the sensitivity of the model to create an optimal model. Apart from increasing the variety of images, augmentation is also useful for balancing data [9]. It is necessary to balance the data because the data that researchers use are unbalanced between classes.

Train Data

The model produced in this stage will be used to conduct trials at the Testing Data stage. Where all information or data from training results are collected in 1 file called model.h5. The flowchart of our proposed method is shown in Figure 1.

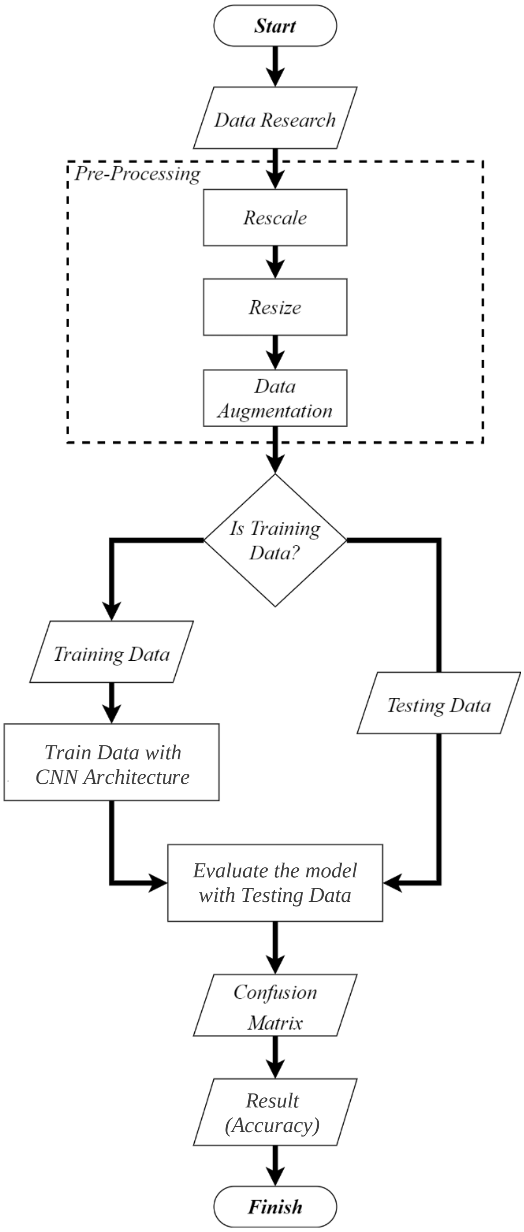


Figure 1. Research stages

Model Evaluation

The evaluation carried out in this study was to analyze the results of the accuracy of implementing the CNN method with the VGG-16 architecture for facial image classification. The model is evaluated using the Confusion Matrix which is assessed from the performance of the model that has been made based on accuracy and to measure performance or performance in the program [12]. The Confusion Matrix is a performance measure for classification problems, where the output can be two or more classes [13]. The Confusion Matrix is a table with 4 different combinations of predicted values and actual values. Hamel in Pratiwi et al. [14], [15] there are four terms that represent the results of the classification process in the Confusion Matrix, namely true positive, true negative, false positive, and false negative, more details can be seen in Table 2.

Table 2. Confusion matrix

		Observed	
		1 True	0 False
Predicted	1 True	True Positive (TP)	False Positive (FP)
	0 False	False Negative (FN)	True Negative (TN)

Based on Table 2, it can be formulated with recall (r), precision (p), and Fscore in Equation 1, as well as accuracy with Equation 2.

$$r = \frac{TP}{TP + FN}\ ,\ \ p = \frac{TP}{TP + FN}\ , \qquad (1)$$

$$\text{accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \qquad (1)$$

From Equation 1 it can be interpreted that recall (r) is the number of correct predictions in the positive class and then divided by the number of positives in the study. Precision (p) is the number of correct predictions in the positive class divided by the number classified as positive. While accuracy is the ratio of correct predictions in all calculations in the positive and negative classes [16].

System Planning

The design of this system is made to test the best model that has been produced through several experiments and test whether the system created using the best model is capable of performing image classification. The system created in this study is web-based, and routing is carried out using flask. Flask is a Python framework as well as a library, which functions as a very helpful tool for web development.

RESULT AND DISCUSSION

Data Retrieval

Comparison of Split Data

Split data in this comparison is done by comparing the amount of training data and testing data, which can be seen in more detail in Table 1.

Table 3. Comparison of split data accuracy

	Distribution of test data and training data		
Split data	70:30	80:20	90:10
Accuracy	0.9103	0.9182	0.9162

Based on Table 3, the distribution of the data used is at a data ratio of 80:20, because the percentage of this data has the highest accuracy compared to 70:30 and 90:10.

Epoch comparison

Epoch is a hyperparameter that is intended to determine how many times the system works through all image data. Based on the experiments that have been carried out using different epoch values, different accuracy results were found.

Table 4. Comparison of accuracy for each epoch

	Accuracy for each epoch			
Epoch	20	60	80	100
Accuracy	0,9182	0,9077	0,9135	0,9086

Based on Table 4, the epoch values that produce the highest accuracy are at epoch 20 compared to epoch 60, 80, and 100. So in training the data will be tested using epoch 20.

Pre-Processing Process

Rescale

This stage is to find the smallest value in the image data up to the values 0 and 1, this is done so that the model can process image data. The image is rescaled to (1/255), so all image data will be divided by 255. The point of dividing by 255 is that 255 is the highest value of the grayscale image. Then later the image value that was (0-255) will shift to (0-1), this process can also be referred to as the data normalization process or looking for the simplest value from the image.

Resize

This step is to change the size of each image, why do this so that the dimensions or dimensions of each image are the same, it will also ease the performance of the model for processing data later. In this process, image data that previously had various or unequal sizes (84x103, 128x128, 524x350), will be converted to 100x100 (100, 100) so that they become images of the same size.

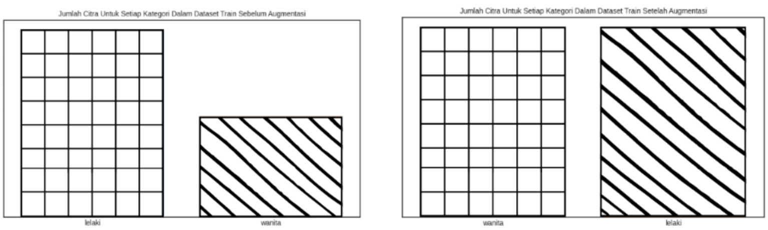
Augmentation

In this stage the image data is processed in such a way as to add variety to the data, including rescaling, rotation, random horizontal and vertical shifts, shears, zooms, and flipping horizontal and vertical images. This augmentation process uses the Keras library, namely ImageDataGenerator whose commands can be seen in Figure 2.

```
train_datagen = ImageDataGenerator(rescale=1./255,
                                   rotation_range = 40,
                                   width_shift_range=0.2,
                                   height_shift_range=0.2,
                                   shear_range=0.2,
                                   zoom_range=0.2,
                                   horizontal_flip=True,
                                   vertical_flip=True,
                                   validation_split=0.2)
```

Figure 2. Augmentation command

The number of datasets after the augmentation is also balanced where the number of datasets for each class is 29,500 images with the final number of datasets being 59,000 images, while the differences in the graphs can be seen in Figure 3 (a) for the initial graph and 3 (b) for the graph after the augmentation.



(a) (b)

Figure 3. (a) Graph before augmentation, (b) Graph after augmentation

Training Data

After completing the design of the VGG-16 Architecture, a model was found that had a total of 155,857 parameters, the details of which can be seen in Figure 4.

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d_4 (Conv2D)           (None, 98, 98, 8)         224

 max_pooling2d_4 (MaxPooling  (None, 49, 49, 8)        0
 2D)

 conv2d_5 (Conv2D)           (None, 47, 47, 16)        1168

 max_pooling2d_5 (MaxPooling  (None, 23, 23, 16)       0
 2D)

 conv2d_6 (Conv2D)           (None, 21, 21, 32)        4640

 max_pooling2d_6 (MaxPooling  (None, 10, 10, 32)       0
 2D)

 conv2d_7 (Conv2D)           (None, 8, 8, 64)          18496

 max_pooling2d_7 (MaxPooling  (None, 4, 4, 64)         0
 2D)

 dropout_1 (Dropout)         (None, 4, 4, 64)          0

 flatten_1 (Flatten)         (None, 1024)              0

 dense_2 (Dense)             (None, 128)               131200

 dropout_2 (Dropout)         (None, 128)               0

 dense_3 (Dense)             (None, 1)                 129

=================================================================
Total params: 155,857
Trainable params: 155,857
Non-trainable params: 0
_________________________________________________________________
```

Figure 4. Modeling results

The parameters resulting from the training process and model design total 155,857 parameters. The layer processing that runs will result in a smaller image size. The smallest size found in the model is 4x4 pixels with 64 filters which will then be processed in a dense layer to turn it into a vector. After the smallest matrix size is converted into a vector, then it will continue to the fc layer which has produced a value of 1024 where this value is a neuron which will later be forwarded to the dense output layer. Then just do the classification based on two categories, the male class and the female class who have been trained.

Model Evaluation

After going through several stages of data processing, the program obtains the best model that has been achieved. This evaluation measurement includes accuracy as shown in Table 5, calculating recall and precision as well as a plot chart that will be displayed from the best model results.

Table 5. Accuracy percentage

	Epoch	20	60	80	100
Before training	Accuracy	0,91	0,90	0,91	0,90
After training	Accuracy	0,94	0,94	0, 92	0,93

Based on Table 5, the best model falls on the model with epoch 20. The researcher tries to calculate precision (p) and recall (r) using Equation 1, and calculates accuracy using Equation 2 based on the values of TP, TN, FP and FN obtained from Confusion calculations Matrix, which is more detailed can be seen in Table 6.

Table 6. Calculation of the confusion matrix epoch 20

		Observed	
		1	0
Predicted	1	5566 (TP)	377 (FP)
	0	314 (FN)	5543 (TN)

$$r = \frac{TP}{TP + FN} \qquad accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$r = \frac{5566}{5566 + 314} \qquad accuracy = \frac{5566 + 5543}{5566 + 5543 + 377 + 314}$$

$$r = \frac{5566}{5880} \qquad accuracy = \frac{11109}{11800}$$

$$r = 0{,}94 \qquad accuracy = 0{,}94$$

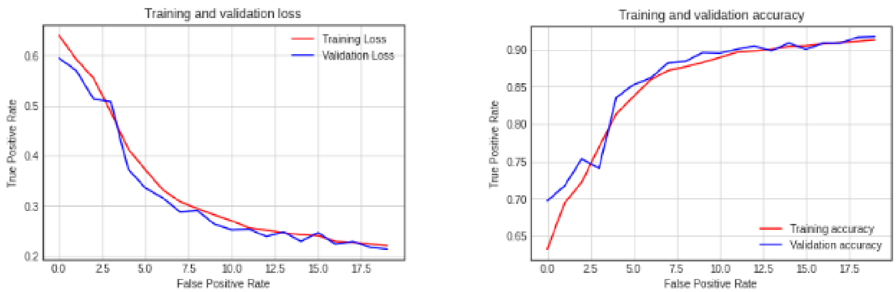
$$p = \frac{TP}{TP + FN}$$

$$p = \frac{5566}{5566 + 314}$$

$$p = \frac{5565}{5943}$$

$$p = 0{,}93$$

In the experimental results using epoch 20, it was found that the smallest accuracy loss was 0.21, then the smallest value validation loss was 0.22 which stopped at epoch 19, the graph of which can be seen in Figure 5 (a). Even though there was an increase in the validation loss from 0.28 to 0.30, after that it could go back down until the end of the training. Which means that during training until the end of the system there are fluctuations and it is normal for this to happen because it means that the system is learning or the training itself is happening. As for Figure 5 (b), the accuracy value reaches 0.91 and the validation accuracy is also 0.91



(a) (b)

Figure 5. (a) Loss of training and validation, (b) Accuracy of training and validation of epoch 20

System Planning

The flow or stages carried out in the system design in Figure 6 are by connecting the model that was made previously with a web framework and then it will be connected to the website with the help of flask. On the Gender Classification web, the user will be asked to enter the image for which the classification test is to be carried out, then the system will classify the image as female or male.

Figure 6. System planning

Before being classified into a female or male class, the system performs face detection on the entered image. After the face is detected, the system will classify the image, and the classification results will appear below the image.

Discussion

From the testing process that has been carried out, a model with the best accuracy is found, namely at an epoch value of 20, which is also proven by the equations that the researchers have calculated. Paying attention to the results of accuracy before (91%) and after training (94%) as well as the results of recall, precision, and Fscore which each produce a value of 94%, this is the best result compared to epoch 60, epoch 80, and epoch 100. Accuracy that has been obtained will be used for comparison with previous studies as shown in Table 7 and Table 8. The CNN method combined with the VGG-16 Architecture is quite good and superior, with an accuracy result obtained of 94%, compared to previous studies which both used the CNN method. Research by Asmara et al. [17] which resulted in an accuracy of 80%, research by Harjoseputro [18] which had an accuracy of 85%, and research by Febriawan [19] which resulted in an accuracy of 92.59%.

Table 7. Comparison of accuracy based on previous research

Name of researcher and year of research	Research title	Research methods	Research result
Asmara et al. (2018)	Gender Classification on Facial Image Using the naive Bayes Method	naive Bayes	80%
Harjoseputro (2018)	Convolutional Neural Network (CNN) for Classifying Javanese Script	Convolutional Neural Network	85%
Febriawan (2022)	Gender Classification on Facial Image Using Convolutional Neural Network and Transfer Learning	Convolutional Neural Network and Transfer Learning	92.59%
Proposed Method	Implementation of Convolutional Neural Network Algorithm Using VGG-16 Architecture for Image Classification on Face Image	Convolutional Neural Network as well as incorporating the VGG-16 Architecture	94%

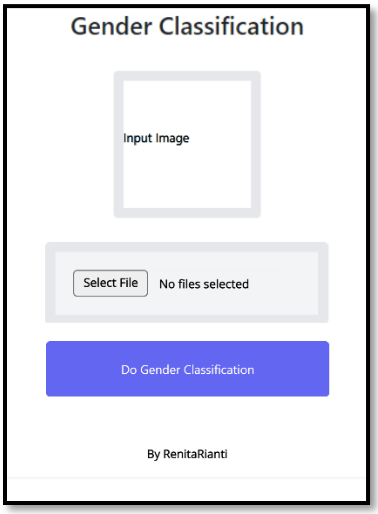
Based on Table 7, a comparison between the proposed method and research Asmara et al. [17] who used the naïve Bayes method obtained an increase of 14%. This is because in his research, the amount of data used is quite small, so the system is not optimal during the learning process. Compared to the proposed method, research Harjoseputro [18], which both used the CNN method, resulted in an increase of 9%. This is because in his research, data testing is done too often, resulting in the model becoming overfitting and difficult to recognize new images. Then the proposed method is compared to research Febriawan [19] which uses the CNN and Transfer Learning methods, the results show an increase of 1.41%. This is because in the autocrop library at the Pre-Processing stage an error was found in the face detection process, as well as changing the RGB color to a gray color, resulting in a less optimal system during the image recognition process. That's where the role of VGG-16 is needed. Not only to produce a good representation of image features, VGG-16 is also capable of recognizing complex image patterns. By getting first place in the competition held by ImageNet, which is a world prestigious competition. It is proven that VGG-16 is indeed the best architecture in its class.

Table 8. Comparison of precision, recall, and fscore with previous research

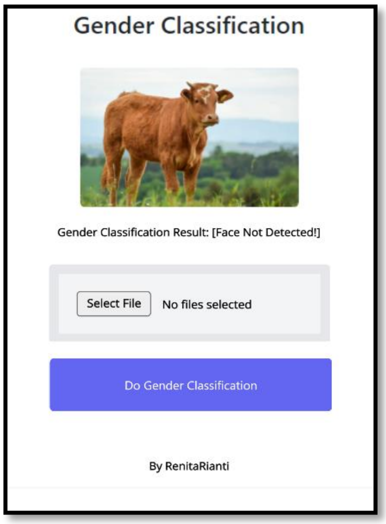
Name of researcher and year of research	Research title	Research methods	Research result
Rohim et al. (2019)	Convolutional Neural Network (CNN) for Image Classification of Traditional Foods	Convolutional Neural Network	73% precision, 69% recall, 69% Fscore
Proposed Method	Implementation of Convolutional Neural Network Algorithm Using VGG-16 Architecture for Image Classification on Face Image	Convolutional Neural Network as well as incorporating the VGG-16 Architecture	94% precision, 94% recall, 94% Fscore

Based on Table 8, the proposed method when compared with the research by Rohim et al. [20] who also used the CNN method found that the proposed method was 21% higher in precision, 25% in recall, and 25% in Fscore compared to previous studies. This is because in his research, there is no VGG-16 Architecture, of course that has an effect on paying attention to the layers contained in the VGG-16 Architecture of course being able to process the data used optimally. So that when training the system is able to capture the information received to the maximum.

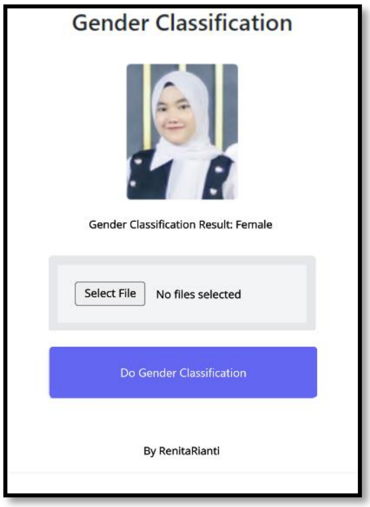
On the Gender Classification web there is a "Select File" button that is used to enter the image you want to test. After the image is entered, the image classification will then be carried out by pressing the "Do Gender Classification" button. Before being classified into a female or male class, the system performs face detection on the entered image. If the system detects that there are no faces in the image, the statement "Face Not Detected" will appear. After the face is detected, the system will classify the image, and the classification results will appear below the image. The initial web appearance can be seen in Figure 7 (a), then the results for face detection can be seen in Figure 7 (b), and the web appearance after classifying can be seen in Figure 7 (c) and Figure 7 (d).



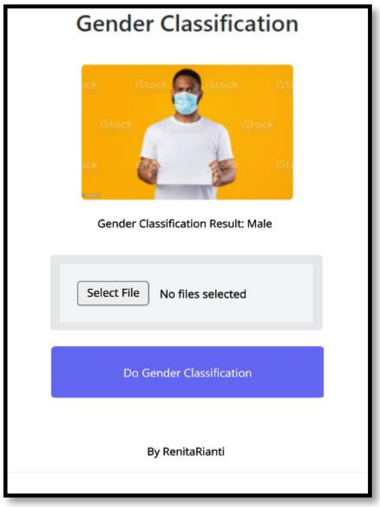
(a)



(b)



(c)



(d)

Figure 7. (a) The initial web appearance (b) The results for face detection (c) Classification result display female (d) Classification result display male

CONCLUSION

The implementation of CNN using the VGG-16 Architecture is quite good, measured by the resulting accuracy reaching 94%. The first implementation process is inputting the facial image dataset, then pre-processing is carried out on each image, namely rescale (1/255), resize (100, 100, 3), data augmentation is also carried out to enrich and balance the image data. Followed by split data, by dividing 80% Train Data and 20% Testing Data. Then conduct training and modeling using the VGG-16 Architecture, to produce the best model which will later be used for testing. Testing is carried out using data in Test Data to test the image classification with the best model. The test was carried out by comparing several epoch values to produce the best accuracy of 94% using an epoch of 20 while the lowest result of the comparison is with an epoch value of 80 and produces an accuracy of 92%. At epoch 60 it also produces a final accuracy of 94%, but the accuracy before training gets 90% accuracy while at epoch 20 it produces 91% accuracy. So the researcher can conclude that the best model is at epoch 20.

REFERENCES

[1] E. Fernando, D. Andwiyan, D. Fitria Murad, D. Touriano, and M. Irsan, "Face recognition system Using Deep Neural Network with Convolutional Neural Networks," *J. Phys. Conf. Ser.*, vol. 1, no. 1, 2019, doi: 10.1088/1742-6596/1235/1/012004.

[2] M. Irsan, S. Ramadhan, and S. A. Murad, "Pendeteksian Wajah Menggunakan Algoritma Convolutional Neural Network Dalam Menghitung Jumlah Mahasiswa," *Kumpul. J. Ilmu Komput.*, vol. 8, no. 3, pp. 279–290, 2021.

[3] H. Sulaiman, Z. Zainuddin, and S. Sahibu, "Sistem Deteksi Wajah Untuk Identifikasi Kehadiran Mahasiswa Dengan Menggunakan Metode Eigenface Pca," *J. Ris. Inform.*, vol. 1, no. 2, pp. 97–106, 2019, doi: 10.34288/jri.v1i2.36.

[4] P. Choirina and R. A. Asmara, "Deteksi Jenis Kelamin Berdasarkan Citra Wajah Jarak Jauh Dengan Metode Haar Cascade Classifier," *J. Inform. Polinema*, vol. 2, no. 4, 2016, doi: 10.33795/jip.v2i4.77.

[5] B. Pribadi and M. Naseer, "Sistem Klasifikasi Jenis Kendaraan Melalui Teknik Olah Citra Digital," *Setrum Sist. Kendali-Tenaga-elektronika-telekomunikasi-komputer*, vol. 3, no. 2, p. 103, 2016, doi: 10.36055/setrum.v3i2.505.

[6] T. Susim and C. Darujati, "Pengolahan Citra untuk Pengenalan Wajah (Face Recognition) Menggunakan OpenCV," *J. Heal. Sains*, vol. 2, no. 3, pp. 534–545, 2021, doi: 10.46799/jsa.v2i3.202.

[7] W. I. Suartika, Y. A. Wijaya, and R. Soelaiman, "Klasifikasi Citra Menggunakan Convolutional Neural Network (Cnn) Pada Caltech 101," *J. Tek. ITS*, vol. 5, no. 1, p. 76, 2016, [Online]. Available: http://repository.its.ac.id/48842/

[8] W. Setiawan, "Perbandingan Arsitektur Convolutional Neural Network Untuk Klasifikasi Fundus," *J. Simantec*, vol. 7, no. 2, pp. 50–51, 2020, doi: 10.21107/simantec.v7i2.6551.

[9] N. U. Oktaviana and Y. Azhar, "Garbage Classification Using Ensemble DenseNet169," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1207–1215, 2021, doi: 10.29207/resti.v5i6.3673.

[10] A. Supriyanto, W. A. Kusuma, and H. Rahmawan, "Klasifikasi Kanker Tumor Payudara Menggunakan Arsitektur Inception-V3 Dan Algoritma Machine Learning," *J. Al-AZHAR Indones. SERI SAINS DAN Teknol.*, vol. 7, no. 3, 2022, doi: 10.36722/sst.v7i3.1284.

[11] L. Perez and J. Wang, "The Effectiveness of Data Augmentation in Image Classification using Deep Learning," *arxiv Comput. Vis.*, vol. 1, no. 2, p. 5, 2017, doi: https://doi.org/10.48550/arXiv.1712.04621.

[12] M. A. Rohman and D. Arifianto, "Penerapan Metode Euclidean Probality dan Confusion Matrix dalam Diagnosa Penyakit Koi," *J. Smart Teknol.*, vol. 2, no. 2, pp. 122–130, 2021, [Online]. Available: file:///C:/Users/dokta/Downloads/4992-14056-2-PB.pdf

[13] M. Heydarian, T. E. Doyle, and R. Samavi, "MLCM: Multi-Label Confusion Matrix," *IEEE Access*, vol. 10, pp. 19083–19095, 2022, doi: 10.1109/Access.2022.3151048.

[14] L. Hamel, *Model Assessment with ROC Curves*. 2011. doi: 10.4018/978-1-60566-010-3.

[15] B. P. Pratiwi, A. S. Handayani, and S. Sarjana, "Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi Wsn Menggunakan Confusion Matrix," *J. Inform. Upgris*, vol. 6, no. 2, pp. 66–75, 2021, doi: 10.26877/jiu.v6i2.6552.

[16] J. Miao and W. Zhu, "Precision–recall curve (PRC) classification trees," *Evol. Intell.*, vol. 15, no. 3, pp. 1545–1569, 2022, doi: 10.1007/s12065-021-00565-2.

[17] R. A. Asmara, B. S. Andjani, U. D. Rosiani, and P. Choirina, "Klasifikasi Jenis Kelamin Pada Citra Wajah Menggunakan Metode Naive Bayes," *J. Inform. Polinema*, vol. 4, no. 3, p. 212, 2018, doi: 10.33795/jip.v4i3.209.

[18] Y. Harjoseputro, "Convolutional Neural Network (Cnn) Untuk Pengklasifikasian Aksara Jawa," *Buana Inform.*, vol. 1, no. 1, 2018.

[19] R. Febriawan, "Klasifikasi Gender Pada Citra Wajah Menggunakan Convolutional Neural Network Dan Transfer Learning," https://repository.uinjkt.ac.id, Jakarta, 2022.

[20] A. Rohim, Y. A. Sari, and Tibyani, "Convolution neural network (cnn) untuk pengklasifikasian citra makanan tradisional," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 7, pp. 7038–7042, 2019, [Online]. Available: http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/5851/2789