Accredited SINTA 2 Ranking
Decree of the Director General of Higher Education, Research, and Technology, No. 158/E/KPT/2021
Validity period from Volume 5 Number 2 of 2021 to Volume 10 Number 1 of 2026

Published online at: **http://jurnal.iaii.or.id**

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 8 No. 5 (2024) 665 - 673 e-ISSN: 2580-0760

Cross-Spectral Cross-Distance Face Recognition via CNN with Image Augmentation Techniques

Nisa Adilla Rahmatika[1*], Maulisa Oktiana[2], Fitri Arnia[3]
[1]Master in Electrical Engineering, Universitas Syiah Kuala, Indonesia
[2,3]Department of Electrical and Computer Engineering Universitas Syiah Kuala, Indonesia
[1]adillanisar@gmail.com , [2]maulisaoktiana@usk.ac.id, [3]f.arnia@usk.ac.id

Abstract

Facial recognition is a critical biometric identification method in modern security systems, yet it faces significant challenges under varying lighting conditions, particularly when dealing with near-infrared (NIR) images, which exhibit reduced illumination compared to visible light (VIS) images. This study aims to evaluate the performance of Convolutional Neural Networks (CNNs) in addressing the Cross-Spectral Cross-Distance (CSCD) challenge, which involves face identification across different spectra (NIR and VIS) and varying distances. Three CNN models—VGG16, ResNet50, and EfficientNetB0—were assessed using a dataset comprising 800 facial images from 100 individuals, captured at four different distances (1m, 60m, 100m, and 150m) and across two wavelengths (NIR and VIS). The Multi-task Cascaded Convolutional Networks (MTCNN) algorithm was employed for face detection, followed by image preprocessing steps including resizing to 224x224 pixels, normalization, and homomorphic filtering. Two distinct data augmentation strategies were applied: one utilizing 10 different augmentation techniques and the other with 4 techniques, trained with a batch size of 32 over 100 epochs. Among the tested models, VGG16 demonstrated superior performance, achieving 100% accuracy in both training and validation phases, with a training loss of 0.55 and a validation loss of 0.612. These findings underscore the robustness of VGG16 in effectively adapting to the CSCD setting and managing variations in both lighting and distance.

Keywords: Face recognition; Cross spectral; Data augmentation technique, Deep learning; CNN architecture

How to Cite: N. A. Rahmatika, F. Arnia, and M. Oktiana, "Cross-Spectral Cross-Distance Face Recognition via CNN with Image Augmentation Techniques", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 8, no. 5, pp. 665 - 673, Oct. 2024.
DOI: https://doi.org/10.29207/resti.v8i5.5929

1. Introduction

Facial recognition is a biometric technique that employs the analysis of facial characteristics to distinguish between individuals. The importance of this technology is growing in a number of fields, including security, surveillance, and access control [1], [2]. Its versatility means that it is being used in a wide range of sectors, from law enforcement to consumer services. This creates a pressing need for reliable and accurate facial recognition systems that can perform well in a variety of conditions. In the real world, facial recognition systems must be able to operate effectively in different lighting and distance conditions, which present significant challenges to their accuracy and reliability [3] - [5]

One of the fundamental concepts in the field of face recognition is that of Cross-Spectral Cross Distance (CSCD). CSCD assesses the capacity of a system to identify faces across diverse lighting conditions and distances. For example, face recognition should remain precise even when images are captured during the day with visible light (VIS) or at night with near-infrared light (NIR), and from both near and far distances. CSCD strives to enhance the resilience and dependability of face recognition systems in the context of lighting and distance variations, a prevalent challenge in surveillance and security applications [3].

The utilization of disparate light spectra, including NIR and VIS, is frequently employed to enhance the efficacy of face recognition in diverse lighting scenarios. NIR images are generated from the heat emitted by objects captured by the camera, enabling face detection in low-light or even total darkness conditions. In contrast, VIS images are generated from light reflected by objects and captured by the camera, which is commonly used in good lighting conditions. While using these two types

Received: 17-07-2024 | Accepted: 17-10-2024 | Published Online: 28-10-2024

of spectra allows face recognition systems to function well in various lighting conditions, it also poses challenges in data processing and analysis due to the different image characteristics.

In recent years, Convolutional Neural Network (CNN) models have demonstrated considerable efficacy in a range of pattern recognition tasks, including face recognition. CNN models offer the advantage of high-degree accuracy in face recognition through their capacity for deep feature extraction. However, the efficacy of these models under different lighting conditions and at varying distances remains underexplored, particularly in the context of datasets comprising NIR and VIS images [6] - [11]. Moreover, previous studies have often been limited by the availability of data that captures the variability in spectral and distance conditions, highlighting the need for advanced data augmentation techniques to enrich the dataset and enhance model performance. Data augmentation refers to the process of modifying and enriching the dataset with diverse transformations that aim to enhance the dataset's diversity and facilitate more effective facial pattern recognition by the model [12] - [14].

Previous studies have explored the phase-based CSCD approach to face recognition for use in security and surveillance applications. For instance, experiments conducted using the Long-Distance Heterogeneous Face Database (LDHF-DB) with homomorphic filtering for photometric normalization and band-limited phase-only correlation (BLPOC) for image matching have demonstrated that Equal Error Rate (EER) and Genuine Acceptance Rate (GAR) can be optimized under certain conditions [3]. However, these studies were often limited to specific scenarios and did not comprehensively address the combined impact of spectral and distance variability on CNN performance.

In addition, other research efforts have proposed methods such as combining wavelet-based Histogram of Oriented Gradients (HOG) and Local Binary Pattern (LBP) features to improve face recognition at long distances, particularly in low-light conditions [15]. Despite these advancements, there remains a significant gap in understanding how different CNN architectures perform under these challenging conditions, especially when integrated with innovative pre-processing and data augmentation techniques.

While substantial progress has been made in the development of facial recognition systems, there remains a critical gap in the literature regarding the performance of CNN models across varying spectral and distance conditions, particularly in the context of VIS and NIR datasets. Most existing studies have either focused on limited distance ranges or have not fully explored the effects of cross-spectral variations on CNN accuracy. Additionally, there is a lack of comprehensive studies that combine homomorphic filtering with advanced data augmentation techniques to address these challenges.

This research aims to address these gaps by evaluating the performance of three CNN models—ResNet50, EfficientNetB0, and VGG16—in recognizing faces across disparate light spectra (VIS and NIR) and at varying distances. Specifically, the study investigates how these models perform in classifying faces and explores the impact of data augmentation on model performance. Before data augmentation, homomorphic filtering is employed as a pre-processing step to enhance image quality by addressing issues of poor lighting and shadows, resulting in a more consistent image for subsequent processing. Additionally, this research investigates the influence of diverse data augmentation techniques on training and validation accuracy.

By expanding on existing research, this study not only contributes to the understanding of CNN model performance under different lighting conditions but also offers practical solutions that can be implemented in real-world face recognition systems. The findings are expected to provide insights into improving the overall effectiveness and reliability of facial recognition technology in security and surveillance applications, thereby addressing the existing gaps in the literature.

2. Research Methods

The research commenced with the collection of the Long-Distance Heterogeneous Face Database (LDHF-DB) dataset. The datasets were then categorized into two classes and subsequently separated into two parts, namely training and validation datasets. The details of the research stages are illustrated in Figure 1.

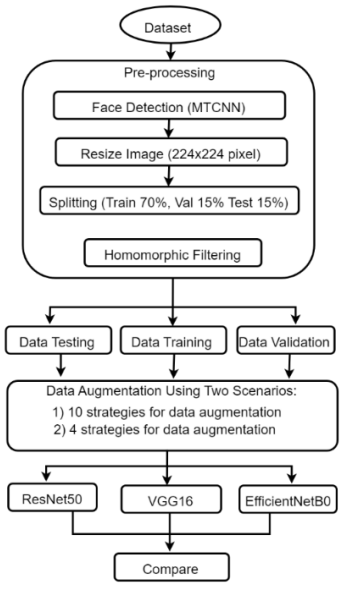

Figure 1. Research Diagram

The research diagram depicts a set of image data (VIS and NIR datasets), which are subjected to a series of

preprocessing stages. These include face detection using MTCNN, image resizing to 224x224 pixels, and the division of the dataset into three distinct subsets: 70% training data, 15% validation data, and 15% testing data. Subsequently, data augmentation was employed to enhance the diversity of the data set and to facilitate the removal of any extraneous noise, thereby improving the overall image quality.

Subsequently, the dataset was divided into two distinct subsets: training data and validation data. The deep learning model was then trained using three different architectures: ResNet50, EfficientNetB0, and VGG16. During the training phase, various parameters, including the learning rate, batch size, and epoch, were adjusted to optimize the training process. Subsequently, the model's performance was evaluated using the validation data, allowing for an assessment of the model's ability to recognize faces in previously unseen data.

2.1 Dataset

The dataset used in this study is derived from the pypi.org/project/bob.db.ldhf database and is designated as LDHF-DB. Table 1 illustrates the distribution of images based on spectrum and distance. Two distinct spectra were employed: visible (VIS) and near-infrared (NIR). Each spectrum comprises samples of images captured at four distinct distances: 1 meter, 60 meters, 100 meters, and 150 meters. For each distance and spectrum, 100 images were captured, with 70 male and 30 female subjects. The total number of images for the visible spectrum was 400, and the total for the near-infrared spectrum was also 400, resulting in a total of 800 images used in this study. Figure 2 illustrates examples of the dataset used for the visible and near-infrared spectra.

Table 1. Number of Datasets in LDHF-DB

Spectrum	Distance				Total
	1m	60m	100m	150m	
VIS	100	100	100	100	400
NIR	100	100	100	100	400
Total					800

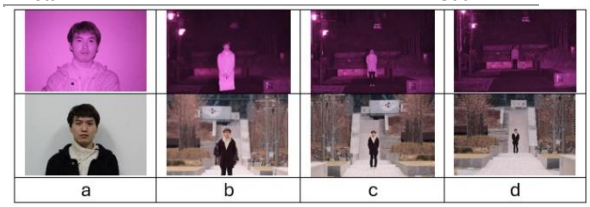

Figure 2. NIR images (top) and VIS images (bottom) at distances of 1m (a), 60m (b), 100m (c), and 150m (d).

2.2 Preprocessing

At this juncture, images of visible and near-infrared wavelengths, captured at distances of 1m, 60m, 100m, and 150m respectively, with 100 images at each distance, underwent a face detection process using MTCNN (Multi-Task Cascaded Convolutional Neural Network) [16] - [18]. This process resulted in 400 images of visible light and 400 images of near-infrared light. The facial detection results were then resized to 224x224 pixels to comply with the requirements of the pre-trained model.

The data was divided using the Split Validation technique. The distribution of data used in the study was 70% for training, 15% for validation, and 15% for testing. This proportion ensures a balanced representation of both data sets [13], as detailed in Table 2.

Table 2. Dataset Distribution

Data	Class	Total
Training	NIR	280
	VIS	280
Validation	NIR	60
	VIS	60
Testing	NIR	60
	VIS	60

2.3 Homomorphic Filtering

Homomorphic filtering is an effective method for enhancing contrast and adjusting brightness in images, particularly when the images suffer from poor illumination. This technique is especially beneficial for images with uneven lighting or prominent shadows, as it works by separating the image's illumination and reflectance components, processing these components separately, and then recombining them. The algorithm for homomorphic filtering typically involves several stages, including Logarithmic Transformation, Discrete Fourier Transformation (DFT), Inverse Discrete Fourier Transformation (IDFT), and Exponential Transformation, with H(u,v) representing the filter applied during the process. Equation 1 can describe the homomorphic filtering process:

$$f(x,y) \rightarrow ln \rightarrow DFT \rightarrow H(u,v) \rightarrow (DFT)^{-1} \rightarrow exp \rightarrow g(x,y) \quad (1)$$

f(x,y) is input image, *ln* is logarithmic transformation, *DFT* is Discrete Fourier Transform, *H(u,v)* is the application of filter, *DFT* is the Inverse Fourier Transform and *exp* stands for exponential transformation.

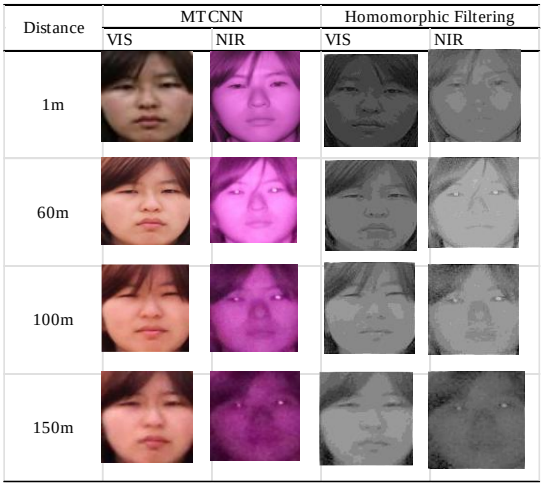

Figure 3 Preprocessing Result

Figure 3 shows the dataset results after the MTCNN process and application of a homomorphic filter.

2.4 Data Augmentation

The augmentation of data is achieved by transforming each image into distinct forms. Generating more images can prevent overfitting, enhance accuracy, and improve the model quality [19] - [24]. This is accomplished through the use of the Image Data Generator, which performs the augmentation process on the dataset [25], [26]. The two principal aspects of the proposed method are evaluated through the assessment of the model results in two scenarios.

The data was subjected to generalization using eight augmentation strategies: rescale, rotation range, width shift range, height shift range, horizontal flip, shear range, zoom range, and brightness range, as well as channel shift range and Gaussian noise.

The data was augmented using four strategies: rotation range, width shift range, horizontal flip, and Gaussian noise.

Table 3. Types of Data Augmentation Parameters

Parameter	Value
Rescale	1./255
rotation_range	40˚
width_shift_range	0.2
height_shift_range	0.2
horizontal_flip	True
shear_range	0.2
zoom_range	0.2
Brightness_range	[0.8, 1.2]
Channel_shift_range	0.2
Gaussian_noise	Noise

Table 3 presents the parameters used for data augmentation, which were employed with eight augmentation strategies to generate a new dataset. These strategies included rescaling, rotation range, width shift range, height shift range, horizontal flip, shear range, zoom range, brightness range, channel shift range, and Gaussian noise. The results of the data augmentation process, which generated 5,040 instances, were used as the training dataset for each of the three models in subsequent analysis. Figure 4 provides illustrative examples of the augmentation techniques employed.

Figure 4. Examples of Data Augmentation on Facial Images

2.5. Model Architecture

The architectural model employed in this study comprises three convolutional neural network (CNN) models, incorporating transfer learning techniques. These include ResNet50 [27], EfficientNetB0 [28], and VGG16 [29]. Table 4 provides a summary of the principal differences between the three CNN models. The models under consideration are VGG16, ResNet50, and EfficientNetB0 [30] - [32].

Table 4. Types of Data Augmentation Parameters

Feature	VGG16	ResNet50	EfficientNetB0
Architecture	Sequential, Simple	Residual Blocks	Compound Scaling
Number of Layers	16 layers	50 layers	236 layers
Main Block	Conv + ReLU + MaxPool	Conv + BatchNorm + ReLU + Shortcut (Residual Block)	MBConv (Mobile Inverted Bottleneck)
Model Size	~528 MB	~98 MB	~20 MB
Depth	Stacked Convolutions	Residual Blocks	MBConv Blocks
Main Advantage	Simplicity and ease of implementation	Addresses degradation in deep networks	High efficiency and accuracy with fewer parameters
Main Disadvantage	Large number of parameters, prone to overfitting	Higher computational complexity	Requires precise tuning for compound scaling
Computation	Slow due to many parameters	Faster than VGG16 with fewer parameters	High efficiency with better computational performance

For further details, please refer to Table 6. VGG16 has a relatively straightforward architectural configuration comprising 16 layers. The primary blocks include a convolutional layer, a rectified linear unit (ReLU), and a max-pooling layer. The model is relatively slow due to the large number of parameters and is susceptible to overfitting. The ResNet50 model features a more intricate architectural design with 50 layers. It employs residual blocks to effectively address potential degradation issues within the network. It offers enhanced speed and efficiency compared to VGG16, although with higher computational complexity. EfficientNetB0 employs composite scaling and MBCConv as its primary blocks, demonstrating

excellent scalability. The hyper-parameters employed throughout the training, validation, and testing phases are detailed in Table 5.

Table 5. Hyperparameters

Hyperparameter	Value
CNN Models	ResNet50, EfficientNetB0, and VGG16
Image Size	224x224 pixels
Dataset Size	5040 training images
Number of Classes	Training, Validation, Testing
Batch Size	32
Training Epoch	100 epochs

3. Results and Discussions

3.1. Preprocessing Result

The facial detection process, conducted using the MTCNN algorithm, yielded a total of 400 VIS images and 340 NIR images. The facial recognition system was unable to detect the faces of six subjects at a distance of 60 meters, nine subjects at a distance of 100 meters, and 45 subjects at a distance of 150 meters. Subsequently, the images were resized to a resolution of 224 x 224 pixels and normalized using homomorphic filtering.

The Image Data Generator was employed to apply eight augmentation techniques to create a new dataset. These techniques included rotation range, width shift range, height shift range, horizontal flip, shear range, zoom range, fill mode, and Gaussian noise at each epoch. Each model was trained using a distinct set of images. The impact of data augmentation on the entire training dataset for the three CNN models (ResNet50, EfficientNetB0, and VGG16) will be investigated.

3.2. Model Performance with Comparison

Once the training and validation procedures were completed and the various augmentation techniques were incorporated, the results of the accuracy assessments were presented in Tables 6 and 7.

Table 6a. Accuracy Results (Scenario I)

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
ResNet50	100%	100%	50%
EfficientNetB0	99,46%	100%	50%
VGG16	100%	99%	55%

Table 6b. Loss Results (Scenario I)

Model	Training Loss	Validation Loss	Testing Loss
ResNet50	0.00093	0.000119	0.705126
EfficientNetB0	0.0124	0.00091	0.768886
VGG16	0.004	0.0305	0.687658

Table 6(a,b) (Scenario I) presents the accuracy and loss results for three convolutional neural network (CNN) models: ResNet50, EfficientNetB0, and VGG16. All three models demonstrated exemplary performance on the training and validation datasets, with training accuracy approaching or reaching 100%. The ResNet50 model exhibited the lowest training loss (0.00093), followed by VGG16 (0.004) and EfficientNetB0 (0.0124). The validation accuracy for ResNet50 and EfficientNetB0 was 100%, while VGG16 achieved the highest testing accuracy of 55%. In contrast, ResNet50 and EfficientNetB0 both exhibited a comparable testing accuracy of 50%. VGG16 also exhibited the lowest testing loss, followed by ResNet50 and then EfficientNetB0. However, there is a discrepancy in the testing loss, with VGG16 exhibiting the lowest loss (0.687658), followed by ResNet50 (0.705126) and EfficientNetB0 (0.768886).

The three models, ResNet50, EfficientNetB0, and VGG16, demonstrated excellent performance on both the training and validation datasets, with almost perfect accuracy. However, a decline in testing accuracy was observed, reaching only 70%, which is indicative of overfitting. Among the three models, ResNet50 exhibited the lowest testing loss, indicating a slight advantage in addressing the testing data compared to the other two models.

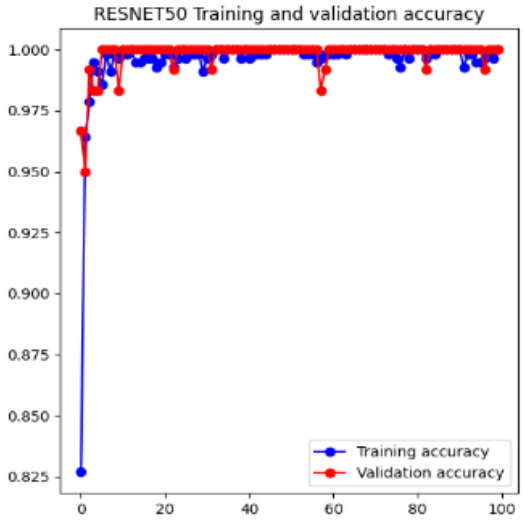

Figure 8. Training and Validation Accuracy Curve of ResNet50

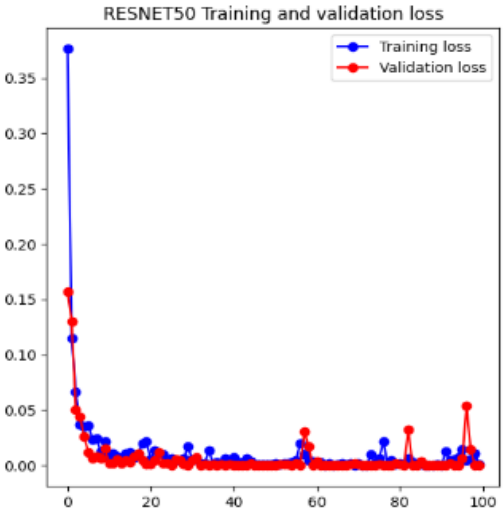

Figure 9. Training and Validation Loss Curve of ResNet50

Figure 8 illustrates a marked increase in training accuracy in the initial epoch, reaching a value of nearly 1.000, indicative of a rapidly learning model. Moreover, the validation accuracy exhibited a rapid increase and subsequent stability at approximately 1.000, indicating effective generalization. Figure 9 shows that the training loss started at approximately 0.35 and exhibited a notable decline towards zero over multiple epochs. The validation loss also showed a rapid decline and remained at a relatively low level, with only minor fluctuations.

The ResNet50 model exhibited excellent performance, demonstrating high accuracy and low loss on both the training and validation datasets. This indicates that the model is both efficient and capable of generalization.

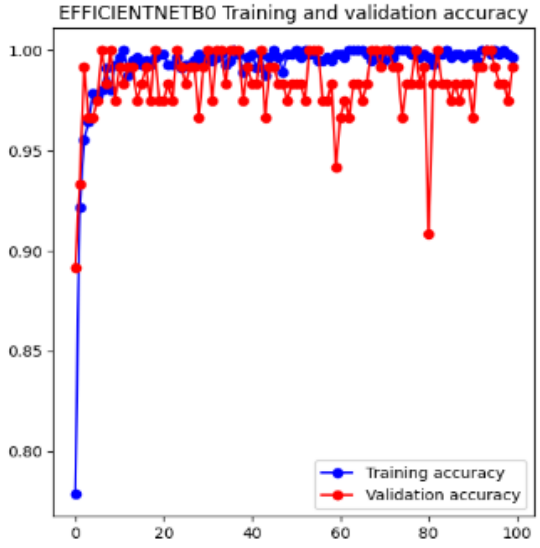

Figure 10. Training and Validation Accuracy Curve of EfficientNetB0

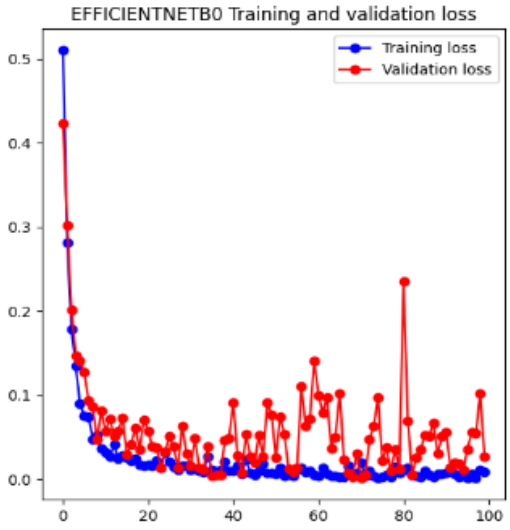

Figure 11. Training and Validation Loss Curve of EfficientNetB0

Figure 10 illustrates that the training and validation accuracy rapidly attain a high and stable value of approximately 0.98 to 1.00. The validation accuracy is also relatively high but exhibits notable fluctuations, particularly after approximately 20 epochs. As shown in Figure 11, the training and validation loss decrease rapidly at the outset and then stabilize, indicating that the model is effectively learning from the training data. However, the validation loss displays considerable fluctuations over epochs, which may indicate overfitting or variation in the validation data.

The EfficientNetB0 model demonstrates exemplary training performance, characterized by high accuracy and minimal loss. However, it is essential to consider the variability observed in the validation performance, which may indicate the necessity for further tuning to enhance generalization.

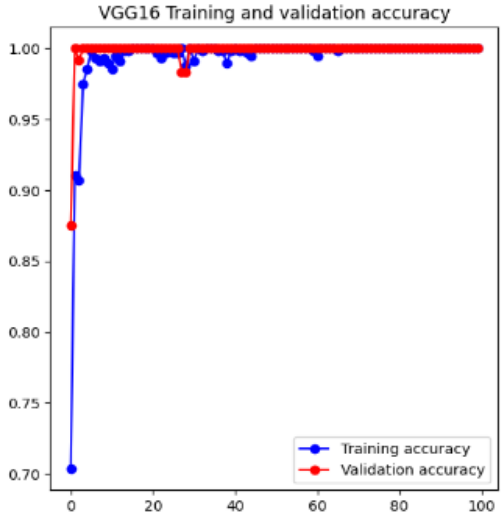

Figure 12. Training and Validation Accuracy Curve of VGG16

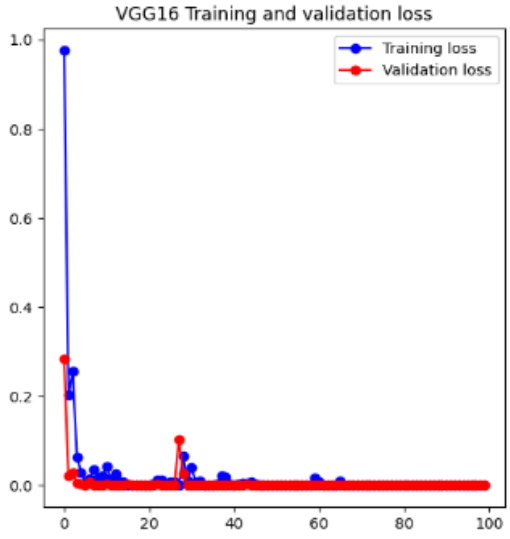

Figure 13. Training and Validation Loss Curve of VGG16

As illustrated in Figure 12, the training and validation accuracy exhibited a rapid increase in the initial epochs, reaching a value of nearly 1.0. This indicates that the model demonstrated excellent and rapid generalization capabilities on the validation data. Figure 13 shows a notable reduction in training and validation loss at the initial epoch, approaching zero and maintaining a low, stable level. This suggests that the model is highly

effective at reducing errors in both datasets. The VGG16 model demonstrated optimal performance, achieving perfect accuracy with minimal loss, indicating excellent performance and robust generalization capabilities.

Table 7a. Accuracy Results (Scenario II)

Model	Training Accuracy	Validation Accuracy	Testing Accuracy
ResNet50	100%	100%	50%
EfficientNetB0	100%	99.17%	50%
VGG16	99.643%	100%	50%

Table 7b. Loss Results (Scenario I)

Model	Training Loss	Validation Loss	Testing Loss
ResNet50	0.00089	0.000153479	0.791316
EfficientNetB0	0.1421	0.008552328	0.829076
VGG16	0.0293	0.000000079	0.612032

Table 7(a,b) (Scenario II) presents the accuracy and loss results for three convolutional neural network (CNN) models: ResNet50, EfficientNetB0, and VGG16. The three models demonstrated exemplary performance on the training and validation datasets, with training accuracy approaching or reaching 100%. The VGG16 model exhibited the lowest training loss (0.000000079), followed by ResNet50 (0.000153479) and EfficientNetB0 (0.008552328). The validation accuracy for ResNet50 and VGG16 was 100%, while that for EfficientNetB0 was slightly lower at 99.17%. In the testing phase, all models demonstrated a consistent testing accuracy of 50%. However, there was a discrepancy in the testing loss data, with VGG16 exhibiting the lowest loss (0.612032), followed by ResNet50 (0.791316) and EfficientNetB0 (0.829076).

The overall performance of the ResNet50, EfficientNetB0, and VGG16 models on the training and validation datasets was excellent, with an almost perfect accuracy rate. However, on the testing dataset, there was a significant decline in performance, with only 50% accuracy, indicating the presence of overfitting. Among the three models, VGG16 exhibited the lowest testing loss, suggesting a slight advantage in handling the testing dataset compared to the other two models.

Figure 14 depicts the performance of the ResNet50 model throughout the training and validation phases. The graph demonstrates that the training accuracy increased rapidly and reached 100% in the initial epochs, maintaining a stable level throughout the remainder of the training process. The validation accuracy also reached 100% with minor fluctuations, indicating that the model was capable of maintaining high performance on unseen data. Figure 15 illustrates that the training loss declined significantly at the outset of training and remained low after several epochs. The validation loss also declined rapidly initially but exhibited greater fluctuations compared to the training loss. Nevertheless, the validation loss remained relatively low overall.

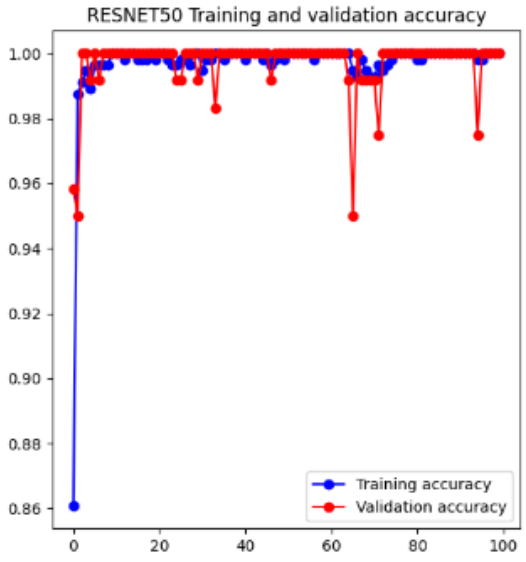

Figure 14. Training and Validation Accuracy Curve of ResNet50

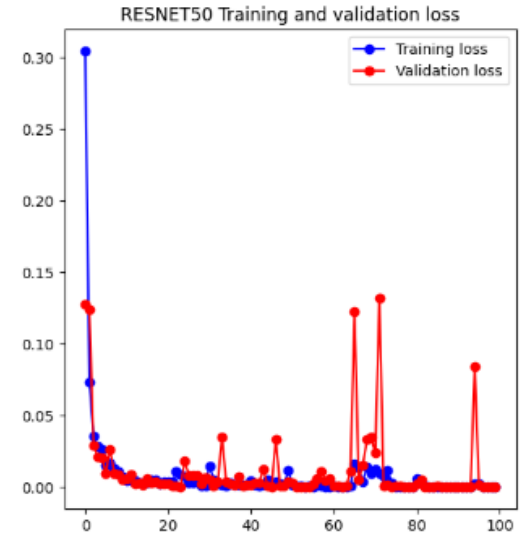

Figure 15. Training and Validation Loss Curve of ResNet50

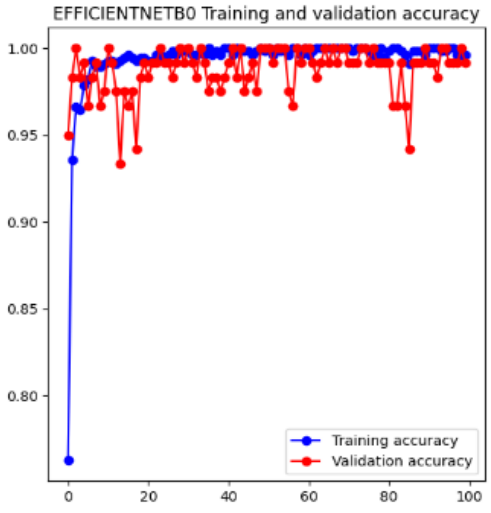

Figure 16. Training and Validation Accuracy Curve of EfficientNetB0

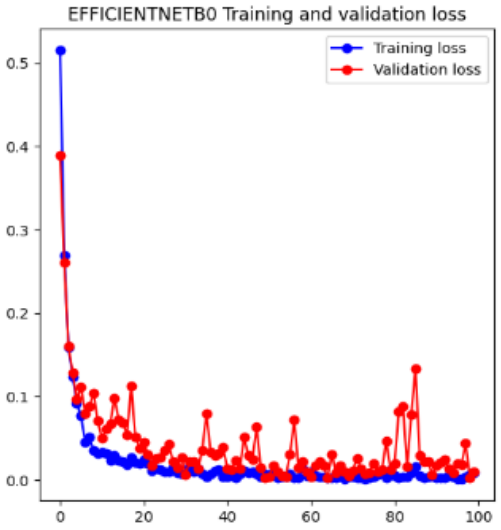

Figure 17. Training and Validation Loss Curve of EfficientNetB0

Figure 16 depicts the performance of the EfficientNetB0 model throughout the training and validation phases. The graph demonstrates that the training accuracy increased rapidly and reached approximately 100% in the initial few epochs, thereafter remaining stable at that level. Validation accuracy also approached 100%, with minor fluctuations across epochs. This suggests that the model can maintain high performance on previously unseen data during the training phase. Figure 17 illustrates that the training loss declines rapidly at the outset and then stabilizes at a minimal value. The validation loss also declines rapidly initially but exhibits greater fluctuations than the training loss. Nevertheless, the overall value of the validation loss remains low.

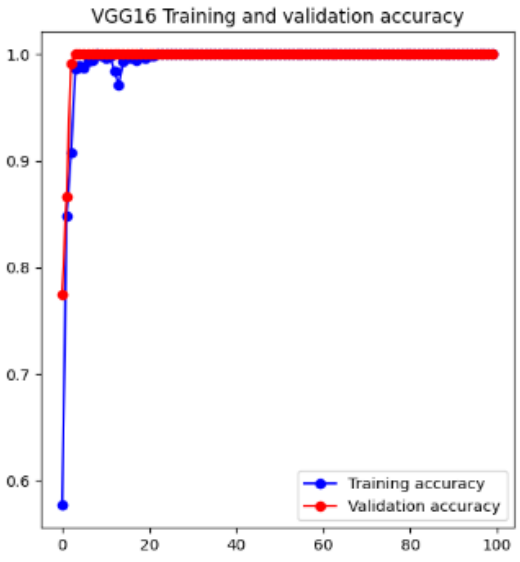

Figure 18. Training and Validation Accuracy Curve of VGG16

Figure 18 depicts the performance of the VGG16 model throughout the training and validation process. The graph demonstrates that the training accuracy increased rapidly, reaching almost 100% in the initial epochs and maintaining a stable level throughout the remainder of the training process. The validation accuracy also reached almost 100% in the initial epochs and remained stable at this level, indicating that the model was able to maintain high performance on unseen data. Figure 19 illustrates that the training loss declined sharply at the outset and then stabilized at a minimal value. Similarly, the validation loss also decreased rapidly initially and remained at a low and stable level throughout the remainder of the training.

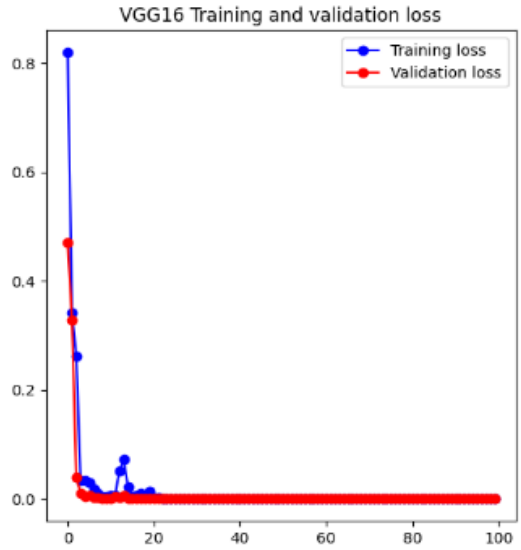

Figure 19. Training and Validation Loss Curve of VGG16

4. Conclusions

This research examines the effectiveness of using Convolutional Neural Networks (CNNs), specifically the VGG16 model, for face recognition in a Cross-Spectral Cross-Distance (CSCD) setting. This setting involves the use of near-infrared (NIR) and visible light spectra, as well as varying distances. The results indicate that VGG16 exhibits superior performance in this context compared to RESNET50 and EFFICIENTB0. This model achieved high training and validation accuracy with low loss, demonstrating strong adaptation to lighting and distance variations in the LDHF-DB dataset. The application of data augmentation techniques, particularly the use of 10 different strategies, proved effective in increasing data diversity and optimizing model performance with a batch size of 32 and 100 epochs. Further research into more advanced data augmentation strategies is essential to improve model generalization and robustness to broader environmental variations. Therefore, future studies are expected to address the challenges of face recognition in heterogeneous environments more effectively.

Acknowledgements

This work was supported by a research grant from the Directorate of Research, Technology, and Community Service.Directorate General of Higher Education, Research and Technology. Ministry of Education, Culture, Research, and Technology In accordance with

the Implementation Contract Master Thesis Research Program Fiscal Year 2024 Number: 094/E5/PG.02.00.PL/2024.

References

[1] K. Guo, S. Wu, and Y. Xu, "Face Recognition Using Both Visible Light Image and Near-infrared Image and a Deep Network," CAAI Transactions on Intelligence Technology, Mar 2017, vol. 2, pp. 39-47.

[2] A. Quagllia and C. M. Epifano, Face Recognition: Methods, Applications and Technology, 2012.

[3] F. Arnia, M. Oktiana, et al., "Homomorphic Filtering and Phase-Based Matching for Cross-Spectral Cross-Distance Face Recognition," *MDPI*, 2021.

[4] M. Oktiana, T. Horiuchi, K. Hirai, K. Saddami, F. Arnia, Y. Away, K. Munadi, "Cross-spectral iris recognition using phase-based matching and homomorphic filtering", *Heliyon*, 2020.

[5] F. Juefei-Xu, D. K. Pal and M. Savvides, "NIR-VIS Heterogeneous Face Recognition via Cross-Spectral Joint Dictionary Learning and Reconstruction," *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Providence, RI, USA, 2015, pp. 141-150, doi: 10.1109/CVPRW.2015.7301308.

[6] J. Lezama, Q. Qiu and G. Sapiro, "Not Afraid of the Dark: NIR-VIS Face Recognition via Cross-Spectral Hallucination and Low-Rank Embedding," *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 6807-6816, doi: 10.1109/CVPR.2017.720.

[7] R. He, J. Cao, L. Song, Z. Sun and T. Tan, "Adversarial Cross-Spectral Face Completion for NIR-VIS Face Recognition," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 5, pp. 1025-1037, 1 May 2020, doi: 10.1109/TPAMI.2019.2961900.

[8] M. Wang and W. Deng, "Deep face recognition: A survey," *Neurocomputing*, vol. 429, pp. 215-244, 2021, doi: 10.1016/j.neucom.2020.10.081.

[9] R. Jiang, C. T. Li, D. Crookes, W. Meng, and C. Rosenberger, "Deep Learning Models for Face Recognition: A Comparative Analysis," *Unsupervised and Semi-Supervised Learning, Eds. Cham, Switzerland: Springer*, 2020, pp. 123-140. doi: 10.1007/978-3-030-32583-1_6.

[10] G. Guo and N. Zhang, "A survey on deep learning based face recognition," *Computer Vision and Image Understanding*, vol. 189, 2019, doi: 10.1016/j.cviu.2019.102805.

[11] J. Aul, A. Roy, S. Mallick, and J. Sil, "A Comparative Study of Deep Learning-Based Face Recognition and Emotion Detection Techniques Using Social Media Customized Cartoon Post," *Computational Intelligence in Pattern Recognition, Springer*, 2023, vol. 725, pp. 389-400. doi: 10.1007/978-981-99-3734-9_33.

[12] A. Kebaili, J. Lapuyade-Lahorgue and S. Ruan, "Deep Learning Approaches for Data Augmentation in Medical Imaging: A Review", *National Library of Medicine*, April 2023, Vol. 9, No.4, doi: https://doi.org/10.3390/jimaging9040081.

[13] E. Shorten, T. M. Khoshgoftaar, "A survey on Image Data Augmentation for Deep Learning," Journal of Big Data, vol. 6, no. 60, pp. 1-48, 2019.

[14] V. Uchoa, K. Aires, R. Veras, A. Paiva, and L. Britto, "Data Augmentation for Face Recognition with CNN Transfer Learning," 2020 International Conference on Systems, Signals and Image Processing (IWSSIP), 2020, pp. 143-148, doi: 10.1109/IWSSIP48289.2020.9145125.

[15] D. Shamia, D. Abraham Chandy, "Intelligent System for Cross-Spectral and Cross-Distance Face Matching," Computers & Electrical Engineering, vol. 67, pp. 228-239, November 2017.

[16] H. Ku and W. Dong, "Face Recognition Based on MTCNN and Convolutional Neural Network", *Frontiers in Signal Processing*, January 2020, vol. 4, no. 1, doi: https://dx.doi.org/10.22606/fsp.2020.41006.

[17] M. Yuan, S. Y. Nikouei, A. Fitwi, Y. Chen and Y. Dong, "Minor Privacy Protection Through Real-time Video Processing at the Edge," *2020 29th International Conference on Computer Communications and Networks (ICCCN)*, Honolulu, HI, USA, 2020, pp. 1-6, doi: 10.1109/ICCCN49398.2020.9209632.

[18] Z. Yang, W. Ge and Z. Zhang, "Face Recognition Based on MTCNN and Integrated Application of FaceNet and LBP Method," *2020 2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM)*, Manchester, United Kingdom, 2020, pp. 95-98, doi: 10.1109/AIAM50918.2020.00024.

[19] W. A. Mustafa, W. Khairunizam, H. Yazid, Z. Ibrahim, A. Shahriman, and Z. M. Razlan, "Image Correction Based on Homomorphic Filtering Approaches: A Study," in Proceedings of the IEEE International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA), pp. 1-5, 2018.

[20] S. Gamini and S. S. Kumar, "Homomorphic filtering for image enhancement based on fractional-order derivative and genetic algorithm," *Computers and Electrical Engineering*, vol. 106, 2023, doi: 10.1016/j.compeleceng.2022.108566.

[21] Á. Chavarín, E. Cuevas, O. Avalos, J. Gálvez and M. Pérez-Cisneros, "Contrast Enhancement in Images by Homomorphic Filtering and Cluster-Chaotic Optimization," in *IEEE Access*, vol. 11, pp. 73803-73822, 2023, doi: 10.1109/ACCESS.2023.3287559.

[22] M. Oktiana, T. Horiuchi, K. Hirai, K. Saddami, F. Arnia, Y. Away, and K. Munadi, "Cross-spectral iris recognition using phase-based matching and homomorphic filtering," Heliyon, vol. 6, no. 10, Oct. 2020.

[23] F. M. Salman and S. S. Abu-Naser, "Classification of Real and Fake Human Faces Using Deep Learning," International Journal of Academic Engineering Research (IJAER), vol. 6, no. 3, pp. 1-14, Mar. 2022.

[24] K. Alomar, H. I. Aysel dan X. Cai, "Data Augmentation in Classification and Segmentation: A Survey and New Strategies", *Journal of Imaging*, 2023, Vol. 9, No. 2, p. 46, doi: https://doi.org/10.3390/jimaging9020046

[25] Francois Chollet, *Deep Learning with Python, Second Edition* , Manning, 2021.

[26] M. Xu, S. Yoon, A. Fuentes, and D. S. Park, "A Comprehensive Survey of Image Augmentation Techniques for Deep Learning," *Pattern Recognition*, vol. 137, pp. 109347, 2023. doi: 10.1016/j.patcog.2023.109347.

[27] S. Lasniari, Jasril, S. Sanjaya, F. Yanto, and M. Affandes, "Klasifikasi Citra Daging Babi dan Daging Sapi Menggunakan Deep Learning Arsitektur ResNet-50 dengan Augmentasi Citra," *Jurnal Penelitian Teknologi dan Rekayasa*, vol. 3, no. 4, pp. 45-58, 2022, doi: 10.1234/jptr.v5i2.1234.

[28] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov and L. -C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 4510-4520, doi: 10.1109/CVPR.2018.00474.

[29] L. Ali, F. Alnajjar, H. A. Jassmi, M. Gocho, W. Khan, and M. A. Serhani, "Performance Evaluation of Deep CNN-Based Crack Detection and Localization Techniques for Concrete Structures," *Sensors*, 2021, vol. 21, no. 5, p. 1688, doi : https://doi.org/10.3390/s21051688.

[30] N. Choudhary, A. Sharma, V. S. Rathore, dan N. Tiwari, "Performance Comparison of ResNet50V2 and VGG16 Models for Feature Extraction in Deep Learning," *SpringerLink*, 2024, pp. 223-229, doi: https://doi.org/10.1007/978-981-99-8031-4_21.

[31] A. Shazia, T. Z. Xuan, J. H. Chuah, et al., "A comparative study of multiple neural network for detection of COVID-19 on chest X-ray," *EURASIP J. Adv. Signal Process*, 2021, no. 50, doi: 10.1186/s13634-021-00755-1.

[32] S. Asif, Q. u. Ain, S. U. R. Khan, et al., "SKINC-NET: an efficient Lightweight Deep Learning Model for Multiclass skin lesion classification in dermoscopic images," *Multimed Tools Appl*, 2024. doi: 10.1007/s11042-024-19489-x.

