The object of study is a solution for early skin disease diagnosis by integrating hybrid deep neural networks – EfficientNetB7 for Classification and YOLOv8 for detection. The system is designed to classify five skin conditions: Melanoma, Basal Cell Carcinoma (BCC), melanoma is a type of skin cancer that originates from melanocytes, the cells that produce skin pigment, Melanocytic Nevi (NV) Melanocytic nevus is a mole or dark spot on the skin formed due to the accumulation of melanocytes, Benign Keratosis-like Lesions (BKL) is a term for a group of skin changes that resemble keratosis but are non-cancerous, and Seborrheic Keratoses and other benign tumors to enhance the health diagnostics. The problem to be solved in this study revolves around improving early and accurate skin disease diagnosis, particularly in resource-limited or underserved areas and the lack of Accessible Diagnostic Tools and Low Efficiency of Current Diagnostic Methods. The study highlights EfficientNetB7's classification accuracy at 94 % and YOLOv8's means average precision (mAP) of 0.812 for detection. This hybrid system processes skin images efficiently, providing classification and detection outcomes with consistent performance in multiple tests. The results demonstrate that the EfficientNetB7 model achieved an accuracy of 94 % on test data, while YOLOv8 delivered a detection performance with a mean average precision (mAP) of 0.812. The web-based system efficiently processed skin images and provided classification and detection outcomes.

Furthermore, integrating EfficientNetB7 and YOLOv8 allowed the skin disease detection system to classify five different diseases and assess malignancy risk. The systems are portable and can be used with minimal setup, making them practical for real-world diagnostic use. The Scope and Practical applications are designed for accessibility in resource-limited settings. The website-based skin disease detection tool provides a user-friendly platform accessible to the public and healthcare providers, especially in underserved areas. Each application's high accuracy and ease of use make them viable aids in early diagnosis, potentially improving healthcare access

Keywords: hybrid deep neural network (HDNN), EfficientNetB7, YOLOv8, skin diseases, coastal communities, health

UDC 681
DOI: 10.15587/1729-4061.2024.313983

DEVELOPING AN EARLY DETECTION MODEL FOR SKIN DISEASES USING A HYBRID DEEP NEURAL NETWORK TO ENHANCE HEALTH INDEPENDENCE IN COASTAL COMMUNITIES

Tengku Henny Febriana Harumy
Corresponding author
Lecturer*
E-mail: hennyharumy@usu.ac.id
Dewi Sartika Br Ginting
Lecturer*
Fuzy Yustika Manik
Lecturer*
Alkhowarizmi
Lecturer
Department of Computer Science
Universitas Muhammadiyah Sumatera Utara
Kapten Muchtar Basri str., 3, Glugur Darat II, Kec. Medan Tim., Medan City, North Sumatera, Indonesia, 20238
*Department of Computer Science
Universitas Sumatera Utara
Dr. T. Mansur str., 9, Padang Bulan, North Sumatera, Indonesia, 20222

Received 27.09.2024
Received in revised form 26.11.2024
Accepted date 11.12.2024
Published date 27.12.2024

How to Cite: *Harumy, T. H. F., Br Ginting, D. S., Manik, F. Y., Alkhowarizmi, A. (2024). Developing an early detection models for skin diseases using hybrid deep neural network to enhance health independence in coastal communities. Eastern-European Journal of Enterprise Technologies, 6 (9 (132)), 71–85. https://doi.org/10.15587/1729-4061.2024.313983*

1. Introduction

Skin diseases are a common health issue faced by Indonesian society, with various types, such as dermatitis, measles, and psoriasis, and are often difficult to diagnose accurately without the help of medical professionals [1]. Limited access to healthcare services, especially in remote areas, makes the need for early diagnosis tools for skin diseases increasingly urgent. Convolutional Neural Networks (CNNs) are artificial neural network (ANN) architecture designed specifically for processing grid-like data, such as images or signals, has demonstrated its effectiveness in image classification, particularly within the realm of dermatology [2]. Several studies have demonstrated the success of using CNNs to classify various skin diseases with high accuracy. However, most previous research has tended to focus on classifying a small number of skin disease types. This study aims to develop and implement an image detection and classification model using Hybrid Deep Neural Network (HDNN) for five different types of skin diseases: Melanoma, Basal Cell Carcinoma (BCC), Melanocytic Nevi (NV), Benign Keratosis-like Lesions (BKL), and Seborrheic Keratoses and other Benign Tumors.

Additionally, the model is designed to detect whether the disease falls into the cancer category to enhance the efficiency and effectiveness of model development. This research applies Transfer Learning Hybrid techniques using EfficientNetB7 and YOLOv8 architectures. The model implementation uses Tensorflow and Ultralytics, with datasets accessed through the Kaggle and Primary Data platform and observation results. The results of this study are expected to improve the ability to diagnose skin diseases early, especially in coastal areas with limited access to healthcare services. Furthermore, the developed model will be integrated into a website as an easily accessible user interface, making it an effective tool in the initial diagnosis process of skin diseases.

Rapid technological and scientific progress continues to tackle complex issues across various domains, yet specific critical topics demand further attention due to their direct impact on health, societal well-being, and sustainable development. One such area is AI-driven diagnostic tools for early disease detection, which addresses the global need for accessible healthcare solutions, particularly for underserved populations in remote regions. Given the significant growth in demand for efficient and cost-effective medical diagnostics, this topic has become central to both public health strategies and technological innovation. As modern lifestyles and population growth contribute to rising health challenges, practical diagnostic tools are increasingly necessary to prevent and manage diseases promptly. For instance, the prevalence of skin diseases and infections, particularly in underserved areas, has highlighted the urgent need for diagnostic technologies that are both accessible and accurate. Although recent advances have introduced new methodologies, significant gaps remain in providing scalable, field-ready solutions. The absence of such tools poses risks for communities lacking access to traditional healthcare facilities, underscoring the pressing need for research and development in this field [1–3].

Research in this area is highly relevant, as the potential applications of new diagnostic tools extend far beyond individual health benefits [4–6]. Outcomes from studies in AI-driven diagnostics can impact public health policies, enable cost-effective screening, and support healthcare providers in delivering timely interventions. Furthermore, advancements in this field are crucial to achieving broader healthcare goals, such as reducing the burden of preventable diseases and promoting health equity [7, 8]. To fully understand the current state and potential of this field, it is essential to analyze existing literature [9]. Further research can strategically address unresolved aspects of AI-driven diagnostics and its applications by identifying what has been accomplished and pinpointing unaddressed areas. Therefore, research on developing and refining AI-based diagnostic systems is timely and essential [4, 10, 11]. It addresses critical healthcare needs, supports the advancement of scientific knowledge, and holds promise for substantial real-world impact in public health and beyond [12].

2. Literature review and problem statement

The paper [11] presents classification using InceptionResNetV2 with Soft-Attention and the new loss function and gives 90 percent accuracy, mean of precision, F1-score, recall, and AUC of 0.81, 0.81, 0.82, and 0.99, respectively. Besides, using MobileNetV3Large combined with Soft-Attention and the new loss function, even though the number of parameters is 11 times less and the number of hidden layers is 4 times less, it achieves an accuracy of 0.86 and 30 times faster diagnosis than InceptionResNetV2. However, there were unresolved issues related to the proposed method, and others still face the problem of badly distinguishing between melanoma and black nevus because, in some cases, the melanoma and the nevus image have the same lesion size and color. The reason for this may be the objective difficulty associated with this research, which is that the running time is quite high, and the proposed method is not lightweight and requires strong memory to analyze it. A way to overcome these difficulties is to improve the system's performance using a graphics processing unit (GPU) and a stable system. This approach was used in [12] to improve the system. However, this problem is invalid for the complex black, big melanoma, or red-black nevus. All this suggests that it is advisable to conduct a study on Some future approaches that can be proposed, which would be to change the type of color to another to fix the same color problem in skin diseases color in the system. The advantages of this paper are that the architecture has 11 times fewer parameters and 4 times fewer hidden layers, achieving an accuracy of 0.86 and a diagnosis 30 times faster compared to InceptionResNetV2. Meanwhile, the disadvantage of this paper is that the model is not yet able to distinguish melanoma and black nevus effectively, as in some cases, the melanoma and nevus images have the same lesion size and color.

The paper [12] introduces a cancer detection model with an accuracy of 87.72 % and a sensitivity of 92.15 %. However, it faces unresolved challenges in distinguishing between benign and malignant skin cancer cases and managing the large number of features generated after extraction. These challenges likely stem from limitations in the dataset and the depth of the features used. A potential solution is to combine deeper feature extraction with additional data informed by human expertise, as demonstrated in [13], where appropriate features were added to improve the model's performance. Despite these limitations, the research highlights how such technologies can address gaps in medical services, enhance disease detection rates, and contribute to better health outcomes, particularly for populations with limited access to healthcare. Future studies could examine the effectiveness of AI-driven diagnostic tools in improving healthcare accessibility, especially in underserved areas. Furthermore, integrating these tools into existing healthcare systems can help identify best practices for implementation, ensuring technological advancements deliver meaningful benefits to communities in need. The strengths of this study lie in its use of ensemble learning and deep learning techniques, which achieve high accuracy and sensitivity through transfer learning and feature selection, resulting in an accuracy of 87.72 % and a sensitivity of 92.15 %. However, its primary drawback is its limitation in distinguishing between different skin cancer types and the excessive number of features after extraction.

The paper [14] introduces a classification/diagnosis model that leverages expert knowledge and represents it semantically in a hierarchical structure. The model achieved an accuracy of 87.25±2.24 % on a test dataset of 1067 images. The semantic summarization of diagnosis scenarios offers potential for further algorithm improvement, supporting the development of future computer-aided decision-making systems. However, unresolved challenges remain, particularly in integrating more human expertise into the algorithms and utilizing result analysis to refine the approach further. Additionally, the study aims to make the system more scalable by handling larger datasets and accommodating various diseases and image types. These challenges stem from the inherent difficulty of practically increasing the dataset size and disease diversity. Addressing these issues requires ongoing efforts, such as expanding datasets and incorporating additional disease types, as demonstrated in [15], which increased the combination of features used. Further analysis is necessary to explore new feature combinations and broaden the range of diseases for diagnosis. This highlights the need for studies on advancing medical diagnostics, ensuring solutions are effective and practical for broader audiences and early detection system implementations. The strengths of this study lie in its ability to diagnose four skin diseases from dermoscopy results, achieving an accuracy of 87.25±2.24 %

on a robust test dataset. However, its primary limitation is the lack of practicality, as the available data constrain the model and cannot accommodate additional disease types.

The paper [16] presents a new framework for automated disease skin diagnosis. Pursuing the goal of improving the performance of existing systems, unresolved issues emerged related to the approach's ability to outperform existing methods in this task, including improving the balanced accuracy (BACC) of the best-compared method from 77 % to 86 %. The approach also attempts to bring more transparency to the decision process." The objective difficulty maybe because it makes relevant research impractical. A way to overcome these difficulties can be to focus on two main directions. The first direction is to improve the predictive performance of the framework, with a particular focus on reducing false negative rates and mitigating class imbalance issues. The second direction will be to perform a large-scale clinical validation of the approach. This approach was used in [17] to improve mitigating class imbalance data. However, improving the system's performance requires more in-depth and accurate observation in data collection. All this suggests that conducting a study at this stage is advisable. The system has great potential for use in actual clinical settings as a training tool for novice dermatologists in datasets. The advantages of this paper are that the model can identify decision options for Dermatoscopic Skin Lesions as a potential solution and provides a new framework for automated melanoma diagnosis. Meanwhile, the disadvantage is that the balanced accuracy (BACC) still needs to be improved.

The paper [18] presents the convolutional and deep-learning neural networks combined with fuzzy clustering or World Cup Optimization algorithms in analyzing dermatoscopic images. This study shows that All of them require an ABCD (asymmetry, border, color, and differential structures) algorithm and its derivates (in combination with the ABCD algorithm or separately). Also, they require a large dataset of dermatoscopic images and optimized estimation parameters to provide high specificity, accuracy, and sensitivity, which can significantly facilitate and improve the level of accuracy and timeliness of diagnostics. However, there were unresolved issues related to improving the approach to optimize the architecture model and image Recognition in the model neural network to improve the level of accuracy and timeliness of diagnostics. This unsolved problem may be because of objective difficulties associated with creating a combination of architectural models. A way to overcome these difficulties is to use neural networks to evaluate features that might be unavailable to the naked human eye. Despite that, there is a need for more datasets to confirm those statements. Machine learning has become a helpful tool in diagnosing skin diseases, especially melanoma. This approach was used in [19] to improve data and hybrid architecture Systems for recognizing Pigmented Skin Lesions with Fusion and Analysis of Heterogeneous. However, a more in-depth study and systematic experimentation are needed to prove the analysis. All this suggests that conducting a study on the Analysis of Early Detection Models using Neural Network specialists in diagnosing skin diseases with Heterogeneous data is advisable. The advantages of this paper are that it finds that the comparison method requires the ABCD algorithm (asymmetry, border, color, and differential structures) and its derivatives (either in combination with the ABCD algorithm or separately) and a large dataset to improve accuracy and timeliness of diagnosis. Meanwhile, the disadvantages of this paper are the suboptimal architecture model and image recognition in the neural network model used.

The paper [20] detects cutaneous cancer with deep learning, which has been challenging because various anatomic structures create curves and shades that confuse the algorithm and can potentially lead to false-positive results. This study shows that to evaluate whether an algorithm can automatically find suspicious areas and predict the likelihood of a malignant lesion. Region-based convolutional neural network technology created 924 538 possible lesions by extracting nodular benign lesions from 182 348 clinical photographs. After manually or automatically annotating these possible lesions based on image findings, convolutional neural networks were trained with 1 106 886 image crops to locate and diagnose cancer. However, there were unresolved issues related to improving the approach comparing benign and malignant skin diseases annotated with their diagnoses and locations, assessing the algorithm's performance on individuals of different races/ethnicities, and additional validation with malignant melanoma. Further, a comparison with the thoroughly mined model is made. This unsolved problem may be because of objective difficulties in obtaining skin disease datasets, both benign and malignant lesions that have been annotated, and their layout locations. A way to overcome these difficulties can be to measure the algorithm's performance, compare the algorithm with existing algorithms, and conduct testing with several conditions and validation. This approach was used in [21] with classification with variable Nonlinear validation and Activation Functions. However, the models have been trained in only six different nonlinear activation functions, allowing to train the model using other nonlinear functions. All this suggests that it is advisable to carry out validation and testing using various validations of the obtained datasets so that the model quality improves. The advantage of this paper is that the model can automatically evaluate and identify suspicious areas and predict the likelihood of a lesion being malignant. The region-based convolutional neural network technology created 924,538 possible lesions by extracting nodular benign lesions from 182,348 clinical photos. Meanwhile, the disadvantages of this paper are that the model is not yet optimal in diagnosing and locating lesions, as well as evaluating the algorithm's performance on individuals from different races/ethnicities, and additional validation with malignant melanoma is needed.

The paper [22] studies an ensemble approach for melanoma classification, starting with using U-net to segment lesion areas in images. This segmentation generates lesion masks that help resize images, focusing on the lesions. Next, five advanced classification models are employed to analyze dermoscopy images, incorporating squeeze-excitation blocks (SE blocks) to enhance the extraction of critical features. Finally, a novel ensemble network is proposed to effectively combine the classification outputs from these five models. However, the approach faces limitations in developing more effective classification methods that account for the unique characteristics of dermoscopy images and their interrelationships. These challenges are primarily attributed to dataset-related difficulties, particularly in preprocessing, as experimental findings show that segmentation significantly boosts classification accuracy. Overcoming these limitations requires improving preprocessing techniques for segmented dataset images and optimizing training and inference models for efficiency. For instance, [23] demonstrated a practical approach by pruning and expanding neural networks to improve computational efficiency. However, precision is prioritized in medical diagnosis

over computational costs, and the proposed cascaded model requires additional computations due to its feature extraction step, making it more resource-intensive than standard ConvNet models. This model suggests that expanding the dataset and refining training processes with advanced functionalities are crucial for improving outcomes.

Additionally, further research on early detection models for skin diseases is highly recommended. The main strength of this study is its ability to enhance feature extraction and propose an ensemble network that integrates results from multiple classification models. On the other hand, its drawbacks include a limited focus on the characteristics of dermoscopy images and their class relationships, mainly due to dataset challenges.

Despite advances in deep neural network learning techniques for skin disease classification, critical gaps remain in classifying similar skin disease types. This challenge is compounded by dataset diversity and size limitations, leading to model generalization and performance issues across demographic groups. Furthermore, integrating human knowledge into these models is often inadequate, resulting in suboptimal decision-making processes. Furthermore, the complexity and computational demands of existing models hinder their practical standardization in clinical settings, while the high dimensionality of the feature set constrains the training model and increases the risk of overfitting.

3. The aim and objective of the study

The study aims to develop and determine an effective early detection model for skin diseases using a hybrid deep neural network. This paper will enhance health independence in coastal communities by enabling early diagnosis and intervention, especially in supported healthcare providers and the public in underserved areas, enabling timely interventions and potentially improving health outcomes.

To achieve this aim, the following objectives are accomplished:

– to collect and analyze data on skin disease prevalence in coastal communities to ensure the model's relevance and specificity;

– to determine the step for developing a hybrid deep neural network model tailored to detect common skin diseases accurately in coastal populations;

– to evaluate the effectiveness of the model in a real-world setting, assessing its accuracy, ease of use, and impact on community health outcomes;

– to develop and determine user-friendly tools that allow healthcare workers and community members to utilize the model effectively for early skin disease detection.

4. Materials and methods

The object of the study is the skin disease classification with a particular focus on the differences between melanoma, Basal Cell Carcinoma, Melanocytic Nevi, Benign Keratosis-like Lesions, Seborrheic Keratoses, and Benign Tumors. The subject of the study is the development and evaluation of a skin disease classification system that utilizes the Hybrid Deep Neural Network technique to differentiate and classify various types of skin diseases in coastal areas.

The central hypothesis of this study is that an integrated skin lesion classification system, which combines neural network algorithms with human expert knowledge and utilizes a diverse and balanced dataset, will significantly help improve the efficiency in distinguishing between melanoma and benign skin lesions compared to current models.

The assumptions made in the study are:

– data quality and diversity. It is assumed that the dataset used for training and testing the classification system is diverse and representative of the various skin conditions and demographic characteristics found in coastal areas;

– model generalization. It is assumed that the developed HDNN model will generalize well to new, unseen data, maintaining high accuracy in real-world applications;

– effectiveness of expert knowledge. It is assumed that incorporating human expert knowledge into the classification process will enhance the model's decision-making capabilities and improve diagnostic performance;

– feature relevance. It is assumed that the features selected for the classification process are relevant and contribute meaningfully to the differentiation of skin lesions.

Simplifications adopted in the study are:

– focus on specific skin conditions. This study addressed the classification task by concentrating on a small number of skin conditions rather than attempting to classify all possible skin lesions;

– controlled data environment. This study was able to ensure a controlled environment for data collection, where consistency of dermatoscopic image quality was ensured, particularly in coastal areas, thereby minimizing variability in image quality;

– limited demographic coverage. This study may have initially focused on a specific demographic group in a coastal area, with plans to expand the analysis to a more diverse population in future studies;

– assumption of uniformity in image acquisition. It was assumed that the images used in this study were acquired using the same technique and equipment, which may not account for variations in imaging protocols.

In this research, several stages will be carried out, as seen in Fig. 1.

This study developed an application system to classify and detect skin diseases using the Hybrid Deep Neural Network combination EfficientNetB7 and YOLOv8 methods. Furthermore, there are several steps taken as follows:

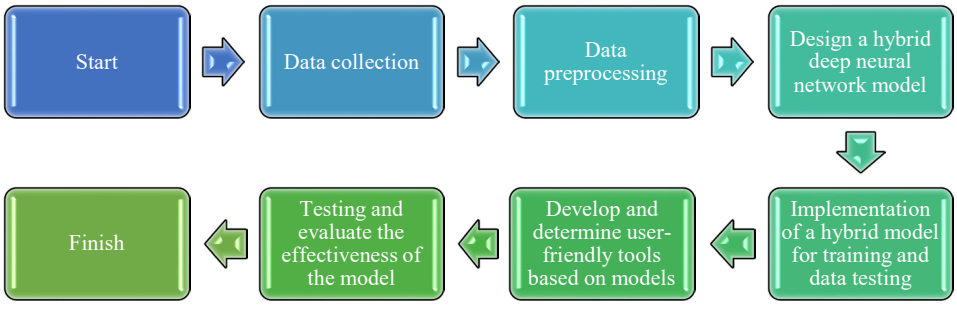

Fig. 1. Methodology research

1. Data Collection. The dataset used for classifying and detecting skin diseases was obtained from the Kaggle platform, and data was directly collected from the research location in the coastal area in the form of image data. Field data was collected through direct photography of individuals in the coastal area.

2. Data Preprocessing. This stage is important in developing a model to classify and detect skin diseases. The image data taken from the Skin Diseases Image Dataset is imported into the programming environment. This data includes images from five previously selected skin disease classes. Next, data annotation is performed, where several image samples from the loaded dataset are selected and annotated using the Roboflow platform. The annotation process involves adding bounding boxes and labels to the images to define specific areas indicating skin or non-skin diseases. Then, image resizing is performed, where each image is resized to consistent dimensions to ensure all images have the same size. The chosen size corresponds to the input requirements of the EfficientNetB7 and YOLOv8 models: 224×224 pixels for EfficientNetB7 and 640×640 pixels for YOLOv8. Next, normalization is carried out to transform the pixel values of the images into the range of [0, 1]. The next step is data augmentation to increase the variation in the dataset and help reduce overfitting, using techniques such as Rotation Range, Width Shift Range, Height Shift Range, Shear Range, Zoom Range, Horizontal Flip, and Fill Mode. Finally, the data is split into Training, Validation, and Testing datasets.

3. Design a hybrid deep neural network:

– input layer: adjusted to the dimensions required by EfficientNetB7 to accept images at a specific resolution;

– pre-trained model usage: utilizing the EfficientNetB7 model on the ImageNet dataset to leverage strong feature representations;

– layer adjustment: adding classification layers such as dense layers with appropriate activation functions, such as ReLU for non-linearity;

– regularization: adding regularization techniques such as dropout to prevent overfitting in the model;

– output layer: using a softmax layer to generate class probability distributions;

– compile: the model is then compiled using the Adam optimizer, the categorical_crossentropy loss function, and the evaluation metrics mentioned earlier;

– dataset configuration setup: prepare the data. yaml configuration file, which defines the path to the training, validation, and testing data directories;

– YOLOv8 model initialization: load the appropriate YOLOv8 model, such as YOLOv8m (medium), which can be used for object detection.

4. Implementation of a Hybrid Model for Training and Data Testing:

– model initialization: the YOLOv8 model is initialized with predefined parameters. This model includes selecting the pre-trained YOLOv8m (medium) model as the starting point for training;

– hyperparameter settings mode: set the mode to "train" to begin the model training. These parameters direct YOLOv8 to run the training on the prepared dataset;

– learning rate: 0.0001, this affects how large the steps the model takes in updating weights during training;

– patience: 10, this determines the number of epochs to go through without significant improvement in the monitored metric (usually loss) before reducing the learning rate or stopping the training;

– batch size: 16; choose the optimal batch size for training the detection model. Larger batch sizes can help the model learn object detection better but require more memory. The batch size is typically adjusted based on GPU capacity and the dataset;

– Number of epochs: 40, the number of complete iterations through the entire dataset. Ensure the number of epochs is sufficient for the model to learn from the data without overfitting;

– dropout: 0.15, apply dropout to reduce the risk of overfitting. Dropout prevents the model from becoming too reliant on specific features by randomly turning off neurons during training;

– model storage: the trained model is saved for use in the next stage, which implements the website-based detection and classification system.

5. Develop and determine user-friendly tools based on models.

In the design of tools based on the model, "hardware" refers to the physical components of the machine that form the system running the program. The components for input, processing, output, and application experiments form the hardware system, which includes 8 GB DDR4 RAM, 2 GB DDR5 Nvidia VGA, 128 GB SSD, and a 500GB hard drive. Testing and evaluating the effectiveness of the model. After the model is trained, the next step is to evaluate it to ensure its performance. This model evaluation is carried out during the validation and testing processes using the validation and testing datasets: 80 % of the data for training (4000 data), 10 % for validation (500 data), and 10 % for testing (500 data). 70 % of the data for training (3500 data), 15 % for validation (750 data), and 15 % for testing (750 data). 50 % of the data for training (2500 data), 25 % for validation (1250 data), and 25 % for testing (1250 data).

5. Results of research of early detection models for skin diseases using hybrid deep neural network

5. 1. Data on skin disease prevalence in coastal communities to ensure the model's relevance and specificity

The dataset for this study was sourced from a combination of the Kaggle and Observation data in the Coastal Area. Platform focusing on five distinct classes of skin diseases relevant to this research:

– melanoma: the most dangerous type of skin cancer occurring in melanin-producing cells;

– basal cell carcinoma (BCC): a common type of skin cancer often caused by excessive sun exposure;

– melanocytic nevi (NV): small spots or generally benign moles;

– Benign keratosis lesion (BKL): typically, harmless skin lesions resembling keratosis;

– seborrheic keratoses and other benign tumors: benign skin tumors commonly found in the elderly.

The dataset comprises 5000 images for the classification task, with 1000 images allocated to each class. This balanced distribution ensures the classification model is trained uniformly across all skin disease categories. The breakdown of images per class is shown in Table 1.

The dataset includes 3000 images categorized into cancerous and non-cancerous groups for the object detection task,

with 1500 images allocated for each class. This dataset ensures a balanced dataset for training the detection model. The distribution of images for the detection task is detailed in Table 2.

Table 1

Classification data

Class	Number of images
Melanoma	1000
Basal cell carcinoma (BCC)	1000
Melanocytic nevi (NV)	1000
Benign keratosis-like lesions (BKL)	1000
Seborrheic keratoses and benign tumors	1000
Total	5000

Table 2

Detection data with Yolo

Class	Number of images
Cancer	1500
Non-cancer	1500
Total	3000

Data preprocessing. The data preprocessing steps are crucial for optimizing the model's training and evaluation. The following steps are conducted:

– loading data: importing images from the dataset into the programming environment;

– data annotation: annotating sample images using a platform like Roboflow, which involves labeling and defining bounding boxes for cancerous and non-cancerous areas;

– resizing: adjusting images to a consistent size suitable for the model input;

– normalization: transforming pixel values to a range of [0, 1] for faster model convergence;

– data augmentation: implementing techniques such as rotation, shifting, shearing, zooming, and flipping to increase data variability;

– Split data: divide the dataset into training, validation, and testing sets.

Model Development and Training. The models for classification and detection of skin diseases are built using the following approaches Classification Model (EfficientNetB7):

– installation of necessary packages, for example, Tensorflow;

– input layer adjustment, use of a pre-trained model, and regularization techniques to avoid overfitting;

– compile the model with appropriate optimizers and loss functions;

– detection model (YOLOv8):

1) installation of the Ultralytics package and setup of dataset configurations for training;

2) initializing the YOLOv8 model for object detection;

3) set the model parameters, for example, mode, epochs, dropout, etc.

After going through the data preprocessing stage, the next stage is the classification and detection model development stage. The model development stage uses the Tensorflow library for classification and Ultralytics for detection. The classification model is trained using training and validation data with epochs of 15, then callbacks such as ReduceLROnPlateau and EarlyStopping from TensorFlow. Keras. Callbacks manage the learning rate and stop training early if specific criteria are met. Model training stage for classification and detection. At this stage, the model is trained with different train, test, and validation data sharing scenarios to assess performance comprehensively, namely (8:1:1), (7:1.5:1.5), and (5:2.5:2.5). Evaluation is carried out by assessing model performance through accuracy, precision, recall, F1-score, and mean average precision (mAP) metrics for object detection.

```
Epoch 12/100
250/250 ━━━━━━━━━━━━━━━━━━━━ 124s 488ms/step - F1Score: 0.9287 - Pr
ecision: 0.9370 - Recall: 0.9253 - accuracy: 0.9300 - loss: 0.2858
- val_F1Score: 0.9336 - val_Precision: 0.9393 - val_Recall: 0.9280
- val_accuracy: 0.9340 - val_loss: 0.3235 - learning_rate: 0.0010
Epoch 13/100
250/250 ━━━━━━━━━━━━━━━━━━━━ 124s 488ms/step - F1Score: 0.9311 - Pr
ecision: 0.9343 - Recall: 0.9260 - accuracy: 0.9309 - loss: 0.2727
- val_F1Score: 0.8310 - val_Precision: 0.8424 - val_Recall: 0.8340
- val_accuracy: 0.8400 - val_loss: 0.5597 - learning_rate: 0.0010
Epoch 14/100
250/250 ━━━━━━━━━━━━━━━━━━━━ 124s 487ms/step - F1Score: 0.9397 - Pr
ecision: 0.9471 - Recall: 0.9342 - accuracy: 0.9403 - loss: 0.2660
- val_F1Score: 0.9062 - val_Precision: 0.9131 - val_Recall: 0.9040
- val_accuracy: 0.9040 - val_loss: 0.3535 - learning_rate: 0.0010
Epoch 15/100
250/250 ━━━━━━━━━━━━━━━━━━━━ 0s 471ms/step - F1Score: 0.9546 - Prec
ision: 0.9571 - Recall: 0.9502 - accuracy: 0.9541 - loss: 0.2057
Training stopped as both training and validation accuracy reached 9
5% or above.
250/250 ━━━━━━━━━━━━━━━━━━━━ 124s 487ms/step - F1Score: 0.9546 - Pr
ecision: 0.9571 - Recall: 0.9502 - accuracy: 0.9541 - loss: 0.2056
- val_F1Score: 0.9543 - val_Precision: 0.9540 - val_Recall: 0.9540
- val_accuracy: 0.9540 - val_loss: 0.1696 - learning_rate:
1.0000e-04
Epoch 12/15
219/219 ━━━━━━━━━━━━━━━━━━━━ 114s 514ms/step - F1Score: 0.9631 - Pre
cision: 0.9652 - Recall: 0.9618 - accuracy: 0.9633 - loss: 0.1756 -
val_F1Score: 0.9431 - val_Precision: 0.9439 - val_Recall: 0.9427 -
val_accuracy: 0.9427 - val_loss: 0.2542 - learning_rate: 1.0000e-04
Epoch 13/15
219/219 ━━━━━━━━━━━━━━━━━━━━ 115s 515ms/step - F1Score: 0.9706 - Pr
ecision: 0.9751 - Recall: 0.9686 - accuracy: 0.9710 - loss: 0.1541
- val_F1Score: 0.9445 - val_Precision: 0.9453 - val_Recall: 0.9440
- val_accuracy: 0.9440 - val_loss: 0.2462 - learning_rate: 1.0000e-05
Epoch 14/15
219/219 ━━━━━━━━━━━━━━━━━━━━ 115s 515ms/step - F1Score: 0.9731 - Pre
cision: 0.9747 - Recall: 0.9716 - accuracy: 0.9730 - loss: 0.1464 -
val_F1Score: 0.9470 - val_Precision: 0.9466 - val_Recall: 0.9453 -
val_accuracy: 0.9467 - val_loss: 0.2421 - learning_rate: 1.0000e-05
Epoch 15/15
219/219 ━━━━━━━━━━━━━━━━━━━━ 115s 515ms/step - F1Score: 0.9750 - Pre
cision: 0.9778 - Recall: 0.9730 - accuracy: 0.9753 - loss: 0.1363 -
val_F1Score: 0.9444 - val_Precision: 0.9452 - val_Recall: 0.9427 -
val_accuracy: 0.9440 - val_loss: 0.2431 - learning_rate: 1.0000e-05
Epoch 12/15
157/157 ━━━━━━━━━━━━━━━━━━━━ 84s 524ms/step - F1Score: 0.9537 - Pre
cision: 0.9577 - Recall: 0.9460 - accuracy: 0.9533 - loss: 0.2319 -
val_F1Score: 0.9321 - val_Precision: 0.9311 - val_Recall: 0.9296 -
val_accuracy: 0.9312 - val_loss: 0.3030 - learning_rate: 1.0000e-04
Epoch 13/15
157/157 ━━━━━━━━━━━━━━━━━━━━ 142s 527ms/step - F1Score: 0.9650 - Pr
ecision: 0.9658 - Recall: 0.9612 - accuracy: 0.9648 - loss: 0.1916
- val_F1Score: 0.9359 - val_Precision: 0.9358 - val_Recall: 0.9336
- val_accuracy: 0.9352 - val_loss: 0.2896 - learning_rate: 1.0000e-04
Epoch 14/15
157/157 ━━━━━━━━━━━━━━━━━━━━ 84s 525ms/step - F1Score: 0.9696 - Pre
cision: 0.9701 - Recall: 0.9663 - accuracy: 0.9699 - loss: 0.1824 -
val_F1Score: 0.9320 - val_Precision: 0.9318 - val_Recall: 0.9296 -
val_accuracy: 0.9312 - val_loss: 0.2970 - learning_rate: 1.0000e-05
Epoch 15/15
157/157 ━━━━━━━━━━━━━━━━━━━━ 84s 525ms/step - F1Score: 0.9709 - Pre
cision: 0.9719 - Recall: 0.9674 - accuracy: 0.9713 - loss: 0.1700 -
val_F1Score: 0.9297 - val_Precision: 0.9303 - val_Recall: 0.9288 -
val_accuracy: 0.9288 - val_loss: 0.3009 - learning_rate: 1.0000e-05
```

Fig. 2. Result model

Based on the results from the model comparison, Model 1 in Fig. 2 (with an 8:1:1 data split) proved to be the best, achieving an accuracy of 95.41 % on the training data, 95.40 % on the validation data and 94 % on the testing data, demonstrating stability and good generalization capability.

5. 2. Determination of the step for developing a hybrid deep neural network model tailored to detect common skin diseases accurately in coastal populations

The step section of the hybrid neural network model designed to detect common skin diseases in coastal populations, there are several stages carried out as follows:

1. Classification model training:

– model initialization: the EfficientNetB7 model that has been built is initialized with initial parameters. This includes the setup of the added classification layers, such as a dense layer with ReLU activation function and an output layer with softmax. Hyperparameter Settings;

– learning rate: 0.001, which can be adjusted during training to ensure good convergence. The learning rate can be iteratively changed to improve model performance;

– batch size: 16; choose an appropriate batch size for training. A larger size increases the stability of training but requires more memory;

– number of epochs: 15, set a sufficient number of epochs for model training.

2. Using callbacks:

– reduce LROn plateau: this callback automatically reduces the learning rate when the monitored metric does not show improvement. This helps the model fine-tune as it approaches convergence;

– model checkpoint: this callback saves the model at specific intervals or when the best performance metric is achieved during training. It ensures researchers can save the best-performing model and avoid losing training progress;

– early stopping: this callback stops training if there is no improvement in the validation metric after a specified number of epochs. This helps prevent overfitting and reduces unnecessary training time;

– model training: the model is then trained using the prepared training and validation data. During training, the model updates its weights to minimize the loss function while monitoring performance metrics;

– model storage: the trained model is saved for use in the next stage, implementing the web-based detection and classification system.

3. Detection model training:

– model initialization: the YOLOv8 model is initialized with predetermined initial parameters. This model includes selecting the pre-trained YOLOv8m (medium) model to be used as the starting point for training;

– hyperparameter settings:

1) mode: set mode to train to start model training. This model directs YOLOv8 to run training on the prepared dataset;

2) learning rate: 0.0001, this affects how large the step the model takes in updating weights during training;

3) patience: 10, this determines the number of epochs that must pass without significant improvement in the monitored metric (usually loss) before the learning rate is reduced or training is stopped;

4) batch size: 16, choose the optimal batch size for training the detection model. A larger batch size can help the model learn object detection better but requires more memory. The batch size is typically adjusted based on GPU capacity and dataset size.

5) the number of epochs is 40, the number of complete iterations through the entire dataset. Ensure the number of epochs is sufficient for the model to learn from the data without overfitting;

6) dropout: 0.15, apply dropout to reduce the risk of overfitting. Dropout prevents the model from relying too much on specific features by randomly turning off neurons during training;

– model storage: the trained model is saved for use in the next stage, which is an implementation in the web-based detection and classification system. To design a hybrid deep neural network model tailored to detect common skin diseases accurately in coastal populations. The detection model was evaluated on two datasets, consisting of 1,500 and 3,000 images, respectively. The model was trained over 40 epochs with a learning rate of 0.0001. The key evaluation metric used to assess detection performance was the mean Average Precision (mAP).

Hybrid deep neural networks YOLOv8 and EfficientNetB7 have similarities in classification, as both utilize convolutional neural networks (CNN). However, their purposes differ. EfficientNetB7 is used for image classification, while YOLOv8 is used for object detection. Both processes include several important stages; feature extraction is extracting features from input images through convolutional layers. For example, there is an image of a skin disease sized 6×6 pixels (36 pixels) with a grayscale color scale. Once the kernel and stride parameters are determined, such as a 3×3 kernel and a stride of 1, the kernel is applied to the image to perform the convolution operation. The stride determines how far the kernel (filter) moves across the input image during the convolution operation. At each convolution step, the kernel performs a dot product operation on the part of the image covered by the kernel, following the stride value until all pixels have been processed. This feature map allows the model to learn different feature representations from the image, which are then used for tasks such as object classification by EfficientNetB7 or object detection by YOLOv8. After pooling, the result is a feature map with lower resolution but with important summarized information. For example, after max pooling, a 2×2 matrix is obtained as a result:

$$\text{Matrix after } max\ pooling = \begin{bmatrix} 3 & 4 \\ 6 & 5 \end{bmatrix},$$

Where this matrix is then further processed depending on the model's task.

Image classification:

1) flattening: next, the 2×2 matrix is flattened into a one-dimensional vector:

Flattened vector=[3, 4, 6, 5];

2) fully connected layers: at this stage, the flattened vector is then fed into the fully connected layers using the formula:

$$z=Wx+b,$$

where:

– W is the weight matrix that connects each element of the flattened vector to the neurons in the fully connected layer;

– x is the flattened vector obtained from the feature map flattening;

– b is the bias added for each neuron in the fully connected layer;

– z is that layer's output, typically followed by an activation function such as softmax. This function determination is also following research [24, 25].

The weights and biases in the artificial neural network are automatically optimized during the training process. For example, since this classification has five classes, the weights W (of size 5×4) and the biases b (of size 5×1) can be given as:

$$W=\begin{bmatrix} 0.1 & -0.2 & 0.3 & 0.4 \\ 0.2 & 0.1 & -0.1 & -0.3 \\ -0.4 & 0.5 & 0.2 & -0.1 \\ 0.3 & -0.4 & 0.1 & 0.2 \\ -0.1 & 0.2 & -0.3 & 0.4 \end{bmatrix} b=\begin{bmatrix} 0.1 \\ -0.1 \\ 0.2 \\ -0.2 \\ 0.1 \end{bmatrix}.$$

Then, it is possible to calculate z value using the formula:

$$z=Wx+b, \quad (1)$$

$$W\cdot x=\begin{bmatrix} 0.1 & -0.2 & 0.3 & 0.4 \\ 0.2 & 0.1 & -0.1 & -0.3 \\ -0.4 & 0.5 & 0.2 & -0.1 \\ 0.3 & -0.4 & 0.1 & 0.2 \\ -0.1 & 0.2 & -0.3 & 0.4 \end{bmatrix} b=\begin{bmatrix} 3 \\ 4 \\ 6 \\ 5 \end{bmatrix},$$

$$W\cdot x=\begin{bmatrix} 0.3 & -0.8 & +1.8 & +2 \\ 0.6 & +0.4 & -0.6 & -1.5 \\ -1.2 & +2 & +1.2 & -0.5 \\ 0.9 & -1.6 & +0.6 & +1 \\ -0.3 & +0.8 & -1.8 & 0.+24 \end{bmatrix} b=\begin{bmatrix} 3.3 \\ -1.1 \\ 1.5 \\ 0.9 \\ 0.7 \end{bmatrix}.$$

Next, add the bias value:

$$z=\begin{bmatrix} 3.3 \\ -1.1 \\ 1.5 \\ 0.9 \\ 0.7 \end{bmatrix}+\begin{bmatrix} 0.1 \\ -0.1 \\ 0.2 \\ -0.2 \\ 0.1 \end{bmatrix}=\begin{bmatrix} 3.4 \\ -1.2 \\ 1.7 \\ 0.7 \\ 0.8 \end{bmatrix}.$$

Each element in the vector z represents each class:
– score for Class 1: $z1$=3.4;
– score for Class 2: $z2$=–1.2;
– score for Class 3: $z3$=1.7;
– score for Class 4: $z4$=0.7;
– score for Class 5: $z5$=0.8;
– activation function: softmax.

To calculate the softmax activation function, it is necessary to convert the raw z scores into probabilities for each class. The softmax function converts the z scores into probability values in the following manner. Score exponentiation:

– calculate the exponent of each score zi normalization;

– divide each score's exponent by the exponents' total sum to get the probability.

The softmax formula can be made as follows:

$$Softmax(zi)=\frac{e^{zi}}{\sum_{j=1}^{n}e^{zj}}, \quad (2)$$

where e^{zi} is the exponent of the i-th score;

$\sum_{j=1}^{n}e^{zj}$ is the sum of the exponents of all scores in the z;

Score z = [3.4, –1.2, 1.7, 0.7, 0.8].

Then:

$$e^{z_1}=e^{3.4}\approx 30.12,$$

$$e^{z_2}=e^{-1.2}\approx 0.30,$$

$$e^{z_3}=e^{1.7}\approx 5.47,$$

$$e^{z_4}=e^{0.7}\approx 2.01,$$

$$e^{z_5}=e^{0.8}\approx 2.23.$$

Total=30.12+0.30+5.47+2.01+2.23=40.13.

Next, let's calculate the probability:

Probability 1=30.12/40.13=0.752;

Probability 2=0.30/40.13=0.0075;

Probability 3=5.47/40.13=0.136;

Probability 4=2.01/40.13=0.050;

Probability 5=2.23/40.13=0.056.

So, after applying the softmax function to the z score, let's get the probability for each class, where the highest probability is in class 1, which is 0.752. Then, the input image is predicted as class 1.

Analysis of the YOLOv8 process.

Feature extraction.

This process is very similar to that in EfficientNetB7. YOLOv8 uses convolutional networks to extract features from the input image. Like EfficientNetB7, the input image is processed through several convolutional layers, resulting in feature maps that represent specific features of the image. At each convolutional layer, the kernel slides over the image and produces output in feature maps that capture various patterns from the image. This function determination is also following research [26–29].

That is a bounding box prediction.

YOLOv8 performs classification and predicts bounding boxes for each object in the image. These bounding boxes are represented by four parameters: (x, y, w, h), where (x, y) is the center point coordinate of the bounding box, and w and h are the width and height of the bounding box. The process of searching for bounding boxes is illustrated in Fig. 3, where the model searches for bounding boxes using anchor boxes of various sizes. At this stage, the model predicts bounding boxes for each object, a confidence score, and the object's class.

Intersection over Union (IoU).

After the bounding boxes are predicted, YOLOv8 calculates the Intersection over Union (IoU) to assess the overlap between the predicted bounding box and the ground truth bounding box. IoU is the ratio between the area of overlap (intersection) of the two bounding boxes and the area of their union. This metric measures how accurately the predicted bounding box compares to the ground truth. The IoU value ranges from 0 to 1, and the higher the value, the better the accuracy of the prediction:

$$\text{IoU}=\frac{\text{Area of Intersection}}{\text{Area of Union}}. \quad (3)$$

Suppose there are two bounding boxes, as shown in Fig. 3.

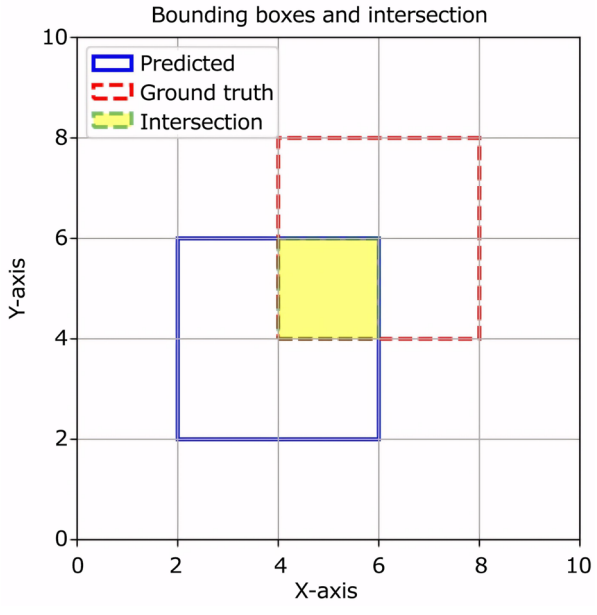

Fig. 3. BBox pred vs. ground truth

Bounding box coordinates:

– predicted bounding box (BB_pred): (2, 2) to (6, 6);

– ground truth bounding box (BB_gt): (4, 4) to (8, 8).

Calculating the area of intersection:

– top-left corner coordinates of the intersection area: (4, 4);

– bottom-right corner coordinates of the intersection area: (6, 6);

– width of the intersection area: 6–4=26–4=26–4=2;

– height of the intersection area: 6–4=26–4=26–4=2;

– area of intersection: 2×2==42\times 2=42×2=4.

Calculating the area of union:

– area of BB_pred: 4×4=164\times 4=164×4=16;

– area of BB_gt: 4×4==164\times 4=164×4=16;

– combined area: 16+16––4=2816+16–4=2816++16–4=28.

– calculating the IoU Value:

IoU=(Area of Intersection)/(Area of Union)=4/28=0.143.

Thus, the IoU value for these bounding boxes is approximately 0.143. Typically, it is possible to set a threshold for IoU to determine whether a predicted bounding box is correct. For example, if the IoU threshold is 0.5, then predictions with IoU≥0.5 are considered correct, while those below are considered incorrect.

Non-maximum suppression (NMS).

YOLO typically predicts multiple bounding boxes and then selects the bounding box with the highest IoU value, a process known as non-maximum suppression (NMS). YOLOv8 uses non-maximum suppression (NMS) techniques to filter overlapping bounding boxes. NMS retains the bounding box with the highest confidence score and removes others if their IoU exceeds a certain threshold.

Output of bounding boxes and class labels.

After applying NMS, YOLOv8 outputs the final filtered bounding boxes and class labels for each bounding box. Each generated bounding box indicates the position and size of the detected object in the image and the class of that object.

Fig. 4 illustrates the training results obtained from the detection model on each dataset. Both figures demonstrate favorable outcomes, where the training loss decreases over epochs while precision, recall, and mAP values increase. Fig. 4 presents the model's performance with the smaller dataset of 1,500 images, indicating steady improvements in the evaluation metrics throughout the training process.

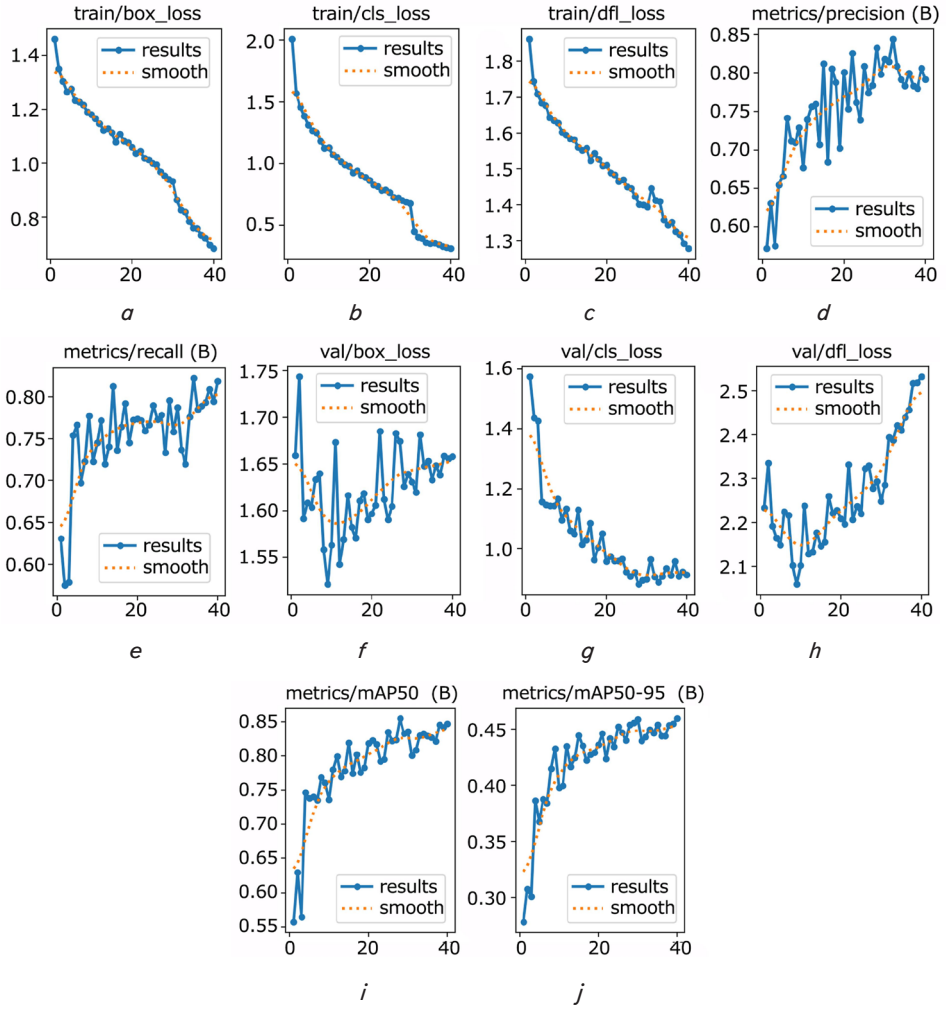

Fig. 4. Graph results for the 1500 images: *a* – train/box_loss training; *b* – train/cls_loss training; *c* – train/dfl_loss training; *d* – metrics/recall(B)_loss training; *e* – metrics/precision(B); *f* – val/box_loss training; *g* – val/cls_loss training; *h* –train/dfl_loss training; *i* – metrics/recall(B)_loss training; *j* – metrics/precision(B)

In contrast, Fig. 5 depicts the model's performance on the larger dataset of 3,000 images, showing an even more pronounced increase in precision, recall, and mAP compared to the smaller dataset. This suggests that the model benefits significantly from a larger dataset, enabling better generalization and more accurate detection results. The numerical comparison of both models is shown in Table 3, which details the MAP achieved by the model on the two datasets.

Table 3

MAP comparison for each detection model

Mean Average Precision (mAP)		
Training	Model 1	Model 2
	1500 images	3000 images
	0.849	0.868
Testing	0.94	0.812

The results in Table 3 demonstrate that the detection model benefits from a larger dataset, as evidenced by the increase in mAP from 0.849 to 0.868. These findings suggest that a more extensive dataset gives the model more representative features, allowing for enhanced generalization in object detection tasks.

In addition to evaluating detection performance using mAP, further analysis was conducted on the lesion sizes derived from the bounding box dimensions in the test data. Specifically, the width and height of each bounding box were extracted to calculate the lesion area, which was then stored in a list.

Thresholds for lesion size classification were determined based on quartiles. The first quartile (Q1, 25th percentile) represents the area below which 25 % of the lesions are smaller, and the third quartile (Q3, 75th percentile) represents the area below which 75 % are smaller. These quartiles were used to categorize lesion sizes as follows:

– small lesions: area<Q1;

– medium lesions: Q1≤area<Q3;

– large lesions: area ≥ Q3.

This quartile-based classification provides a detailed evaluation of the model's detection performance across different lesion sizes, offering insights into its capability to handle objects of varying scales.

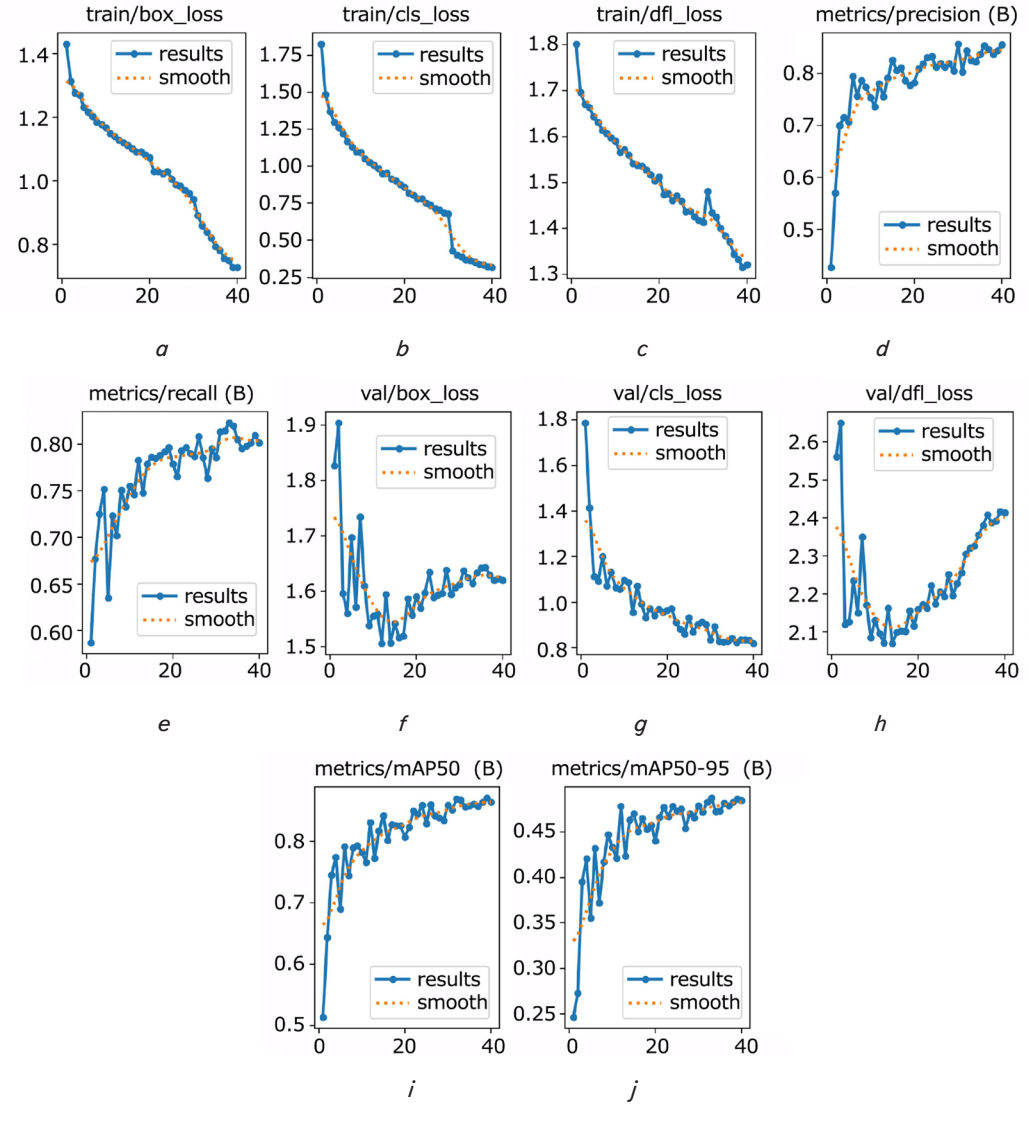

Fig. 5. Graph results for the 3000 images: *a* – train/box_loss training; *b* – train/cls_loss training; *c* – train/dfl_loss training; *d* – metrics/recall(B)_loss training; *e* – metrics/precision(B); *f* – val/box_loss training; *g* – val/cls_loss training; *h* –train/dfl_loss training; *i* – metrics/recall(B)_loss training; *j* – metrics/precision(B)

5. 3. Evaluation of the effectiveness of the model in a real-world setting

The classification model was trained using three different data split scenarios: (8:1:1), (7:1.5:1.5), and (5:2.5:2.5), which correspond to the proportions of data allocated for training, validation, and testing. Early stopping techniques were applied to prevent overfitting during the training process, lasting 15 epochs. Fig. 6–8 show the accuracy over epochs for training and validation across the different data split scenarios, respectively. Evaluate determination is also following research [25–27].

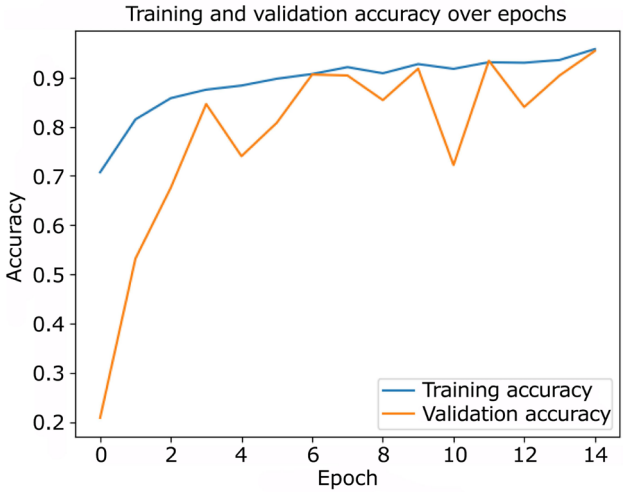

Fig. 6. Training and validation accuracy for scenario 1

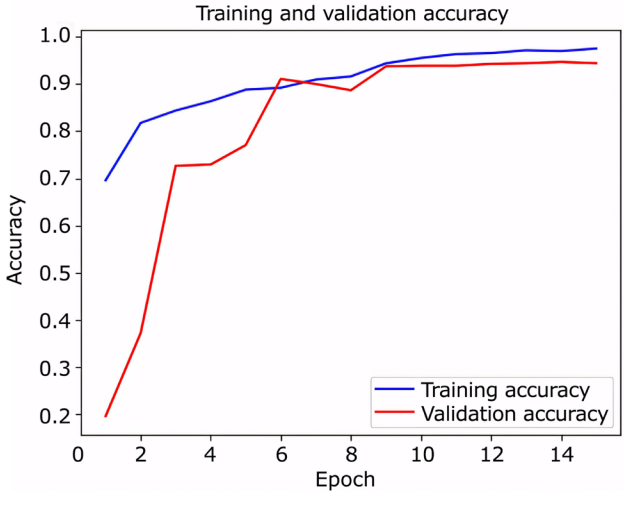

Fig. 7. Training and validation accuracy for scenario 2

The results from the accuracy graphs indicate promising performance across all three scenarios. Despite some fluctuations in the accuracy values throughout the training epochs, the models achieved their highest accuracy at the end of the training process. This trend suggests that the models effectively learned from the data, demonstrating their capacity to classify skin diseases with increasing accuracy as training progressed.

The performance of each model was assessed using metrics such as accuracy, precision, recall, and F1-score, with the results for training, validation, and testing accuracy summarized in Tables 4–7.

Based on the outcomes, Model 1 (8:1:1) achieved the most balanced and stable performance across all phases, with no significant overfitting observed. On the other hand, Models 2 and 3 showed slightly higher training accuracy but experienced a drop in validation performance, indicating potential overfitting.

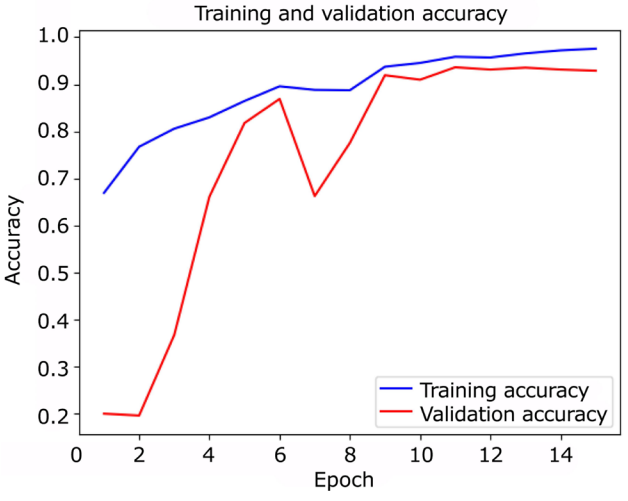

Fig. 8. Training and validation accuracy for scenario 3

Table 4

Accuracy comparison for each classification model

Accuracy			
Training	Model 1	Model 2	Model 3
	(8:1:1)	(7:1.5:1.5)	(5:2.5:2.5)
	0.9541	0.9753	0.9713
Validation	0.9540	0.9440	0.9288
Testing	0.94	0.94	0.94

Table 5

Precision comparison for each classification model

Precision			
Training	Model 1	Model 2	Model 3
	(8:1:1)	(7:1.5:1.5)	(5:2.5:2.5)
	0.9571	0.9778	0.9719
Validation	0.9540	0.9452	0.9303
Testing	0.94	0.94	0.94

Table 6

Recall the comparison for each classification model

Recall			
Training	Model 1	Model 2	Model 3
	(8:1:1)	(7:1.5:1.5)	(5:2.5:2.5)
	0.9502	0.9730	0.9674
Validation	0.9540	0.9427	0.9288
Testing	0.94	0.94	0.94

Table 7

F1-Score comparison for each classification model

F1-Score			
Training	Model 1	Model 2	Model 3
	(8:1:1)	(7:1.5:1.5)	(5:2.5:2.5)
	0.9546	0.9750	0.9709
Validation	0.9543	0.9444	0.9297
Testing	0.94	0.94	0.94

Based on the outcomes, Model 1 (8:1:1) achieved the most balanced and stable performance across all phases, with no significant overfitting observed. On the other hand, Models 2 and 3 showed slightly higher training accuracy but experienced a drop in validation performance, indicating potential overfitting.

5. 4. Development of user-friendly tools

Furthermore, in developing and determining easy-to-use tools, a website-based system was created that can be used by users directly, consisting of several features, including:

1. System implementation.

The system begins with the main page, where users are introduced to the program. As shown in Fig. 9, the user can proceed by clicking the button to begin image classification and detection. This simple interface ensures easy use for anyone interacting with the system for the first time.

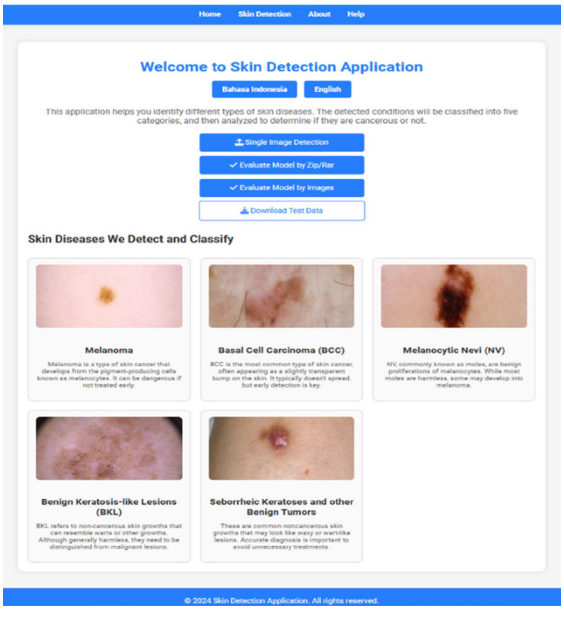

Fig. 9. Main page

In the Skin Detection Application shown in the Fig. 9, several features allow users to upload images of their diseases, such as melanoma, BCC, melanocytic nevi, benign keratosis, and other benign tumors. The system will then detect and classify the type of disease the user has.

2. Prediction page.

Fig.10 shows the implementation of the prediction page that allows the user to upload an image. After the image is uploaded, the website processes it and displays the classification and detection results related to the previously mentioned skin diseases.

Next, in Fig. 10, the prediction page is implemented, allowing users to upload images. After the image is uploaded, the website processes it and displays the classification results along with the detection of the skin diseases previously mentioned.

3. Results page.

After processing the image, the results are displayed on the results page, as shown in Fig. 11. This page provides the predicted skin disease class and a detailed detection output.

The system marks the detected region of interest with a bounding box and predicts whether the condition is cancerous or non-cancerous. This enhances the clarity and interpretability of the results for the users.

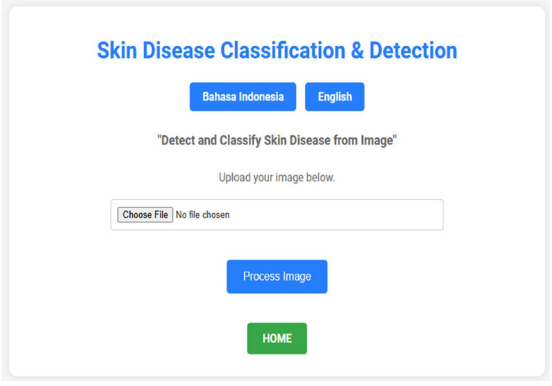

Fig. 10. Prediction page

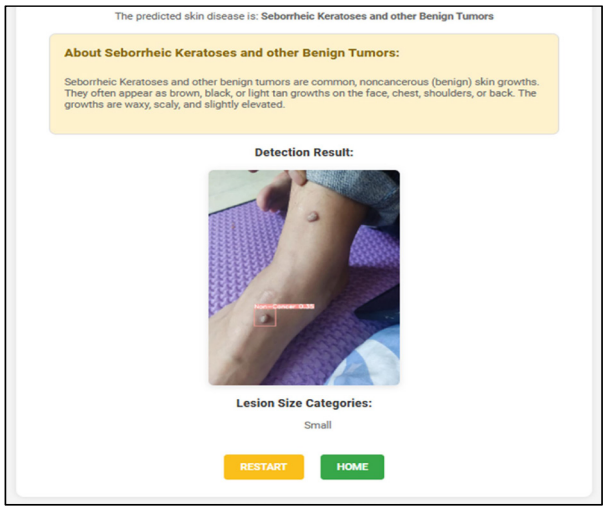

Fig. 11. Classification and detection results page

4. System testing.

Black box testing is a software testing method that evaluates the system's functionality without analyzing its internal architecture or implementation. The primary aim of this testing phase is to ensure that all features in the web-based application, such as image processing, classification, and detection, function as intended. The results of the functional testing are presented in Table 8.

All the features function properly, including Upload Image, Process Image, Home Button, Restart, and other buttons.

Table 8

Functional testing

No.	Feature	Working (Yes/No)	Description
1	Button to program	Yes	If clicked, the user will be directed to the desired program
2	Upload image	Yes	Button to upload an image of the skin disease that the user wants to classify and detect
3	Process image	Yes	Button to process the uploaded image
4	Button home	Yes	Button to return to the main page
5	Button restart	Yes	Button to restart the program
6	Button language	Yes	Button to change the language on the website to Indonesian/English

6. Discussion of the early detection models for skin diseases using a hybrid deep neural network

The hybrid deep neural network model leverages the architectural capabilities of multiple neural networks, including EfficientNetB7 for classification and YOLOv8 for detection, allowing for more effective spatial and temporal feature extraction. The classification and detection results are supported by a comprehensive dataset of around 5000 (Table 2) and 3000 detection data points (Table 2), taken from existing datasets and data collected in the field. In this case, the more significant number of datasets can improve the accuracy rate compared to using 1067 Image data [14]. The performance metrics of the HDNN model show significant progress in accuracy and detection of skin diseases. The overall accuracy rate of 94 % is a noteworthy achievement, especially considering the complexity involved in diagnosing skin conditions that often have overlapping features. The precision, recall, and F1-score metrics strengthen the model's effectiveness, especially in identifying malignant conditions such as melanoma.

The precision rates for melanoma (92 %), BCC (90 %), and NV (93 %) indicate that this model can minimize false positives, which is very important in clinical settings where misdiagnosis can cause anxiety and unnecessary treatment for patients (Fig. 4, 5). From 3 Models (8:1:1), (7:1.5:1.5), and (5:2.5:2.5), The recall rate shows that this model successfully identified a high percentage of actual cases, which is very important for early intervention and treatment (Table 6). The F1 score for various skin diseases reflects a balanced performance, indicating that this model does not reduce precision for recall or vice versa (Table 7). This balance is essential in medical diagnostics; false positives and negatives can have serious consequences. For the research results, it was found that the HDNN model gave better results compared to InceptionResNetV2 with Soft-Attention by 90 % [11] and using a combination of ensemble learning and deep learning by 87.72 % [12], so this model is expected to be an alternative to improve the level of model accuracy.

Furthermore, it provides website visualization to facilitate users in making decisions, which is an advantage compared to previous models [13]. In addition to quantitative metrics, qualitative prediction model analysis provides valuable insights into its practical application. The ability of the hybrid DNN model to accurately identify and localize skin lesions is a significant advantage.

For example, the model successfully highlighted malignant lesions in some test cases while accurately classifying benign conditions. The bounding boxes generated by YOLOv8 not only demonstrate the model's detection capabilities but also provide visual evidence that a web-based system can assist dermatologists in their assessment (Fig. 9). In Model Initialization setup: The YOLOv8 model is initialized with predefined initial parameters. This includes selecting the pre-trained YOLOv8m (medium) model as the starting point for training.

The mode is set to "train" to begin training the model. This directs YOLOv8 to run the training on the prepared dataset. The learning rate is set to 0.0001, which affects how many steps the model takes to update the weights during training. Patience is set to 10, which determines the number of epochs that must pass without significant improvement in the monitored metric (usually loss) before the learning rate is reduced or training is stopped. The batch size is set to 16, the optimal batch size for training the detection model. Larger batch sizes can help the model learn better object detection but require more memory. The number of epochs is 40, which is the number of complete iterations through the entire dataset. Let's ensure that the number of epochs is sufficient for the model to learn from the data without overfitting. A dropout of 0.15 is applied to reduce the risk of overfitting. Dropout prevents the model from relying too much on specific features by randomly deactivating neurons during training.

Determining this parameter is also one of the important indicators for determining the detection results (Fig. 4, 5). The benefit of this combination of models is that it offers a cost-effective alternative to in-hospital or clinic assessments, making it very beneficial for coastal communities. Transfer learning and further refinement improve training efficiency and reduce the need for computational resources (Fig. 10). These qualitative results also confirm the model's potential for real-world applications, especially in settings where access to dermatology expertise is limited. By providing accurate and timely information, the model can empower healthcare providers to make informed decisions regarding patient care. The hybrid DNN model leverages the capabilities of multiple neural network architectures, including EfficientNetB7 for classification and YOLOv8 for detection, allowing for more effective spatial and temporal feature extraction. As a result, this model can identify unique patterns and textures in skin disease images, which is the core of this study and is crucial for accurate diagnosis. This high diagnostic precision helps reduce potential errors and is more efficient (Fig. 11). various models, earlier and more precise identification becomes possible, allowing for faster results even before a thorough examination. In addition, this application offers a cost-effective alternative to in-hospital or clinic assessments, making it very beneficial for coastal communities. Transfer learning and further refinement improve training efficiency and reduce the need for computing resources. Model development is expected to develop datasets that are both image-based and video data to provide better results.

The limitation of this model is that it is developed based on a website, so users must first input image data and then process it. This is one of the shortcomings of implementing this model, and it is expected that mobile-based applications can be developed to be more effective and efficient for further research.

7. Conclusions

1. Collected and analyzed data on skin disease prevalence in coastal communities to ensure the model's relevance and specificity. The combination of the EfficientNetB7 model for classification and YOLOv8 for detection has successfully provided an effective solution for diagnosing five types of skin diseases, including Melanoma, BCC, NV, BKL, and Seborrheic keratoses and other Benign Tumors. Based on the results from the model comparison, Model 1 (with an 8:1:1 data split) proved to be the best, achieving an accuracy of 95.41 % on the training data, 95.40 % on the validation data, and 94 % on the testing data, demonstrating stability and good generalization capability.

2. The steps in developing the hybrid neural network model to detect common skin diseases in coastal populations include the integration of YOLO and a neural network. This hybrid model combines a classification model trained using the ReLU activation function and an output layer with soft-

max activation. Additionally, various callbacks are utilized, including ReduceLROnPlateau, ModelCheckpoint, and EarlyStopping. The detection model is trained with specific hyperparameter settings to achieve better detection results, aiming for a validation accuracy of 0.940 or 94 %.

3. The model's effectiveness was evaluated in a real-world setting, assessing its accuracy, ease of use, and impact on community health outcomes. It is stated that this model does not show significant signs of overfitting, making it superior to Model 2 and Model 3. Meanwhile, in detection, the model comparison indicates that Model 2 (3000 images) is the best model, with a mean average precision (mAP) of 0.812 on the test data, higher than Model 1, which only achieved 0.771. This demonstrates that Model 2 accurately detects skin disease objects in images.

4. User-friendly tools werfe developed that allow healthcare workers and community members to utilize the model effectively for early skin disease detection. Functional testing using black box testing shows that all features on the website work as intended, including image uploading, classification and detection predictions, and the display of prediction results. Furthermore, the model performance testing indicates that the developed system performs well and can be relied upon for classifying and detecting skin diseases.

Conflict of interest

The authors declare that they have no conflict of interest about this research, whether financial, personal, authorship, or otherwise, that could affect the research and its results presented in this paper.

Financing

The study was performed with financial support from Talenta Research grant number 89/UN5.4.10.S/PPM/KP-TALENTA/RB1/2024 Universitas Sumatera Utara.

Data availability

Manuscript has associated data in a data repository.

Use of artificial intelligence

The authors have used artificial intelligence technologies within acceptable limits to provide their own verified data, which is described in the research methodology section.

Acknowledgment

We want to express our deepest gratitude to everyone who contributed to the completion of this project. First and foremost, we would like to thank TALENTA Research and Universitas Sumatera Utara for their financial and logistical support.

References

1. Liopyris, K., Gregoriou, S., Dias, J., Stratigos, A. J. (2022). Artificial Intelligence in Dermatology: Challenges and Perspectives. Dermatology and Therapy, 12 (12), 2637–2651. https://doi.org/10.1007/s13555-022-00833-8
2. He, Z., Marrone, G., Ou, A., Liu, H., Ma, L., Huang, Y. et al. (2020). Factors affecting health-related quality of life in patients with skin disease: cross-sectional results from 8,789 patients with 16 skin diseases. Health and Quality of Life Outcomes, 18 (1). https://doi.org/10.1186/s12955-020-01542-6
3. Prasitpuriprecha, N., Santaweesuk, S., Boonkert, P., Chamnan, P. (2022). Prevalence and DALYs of skin diseases in Ubonratchathani based on real-world national healthcare service data. Scientific Reports, 12 (1). https://doi.org/10.1038/s41598-022-20237-0
4. Paithankar, V., Uprit, I., Bairagi, A., Zoting, H., Katarnavare, A. (2023). Skin Disease Identification using Image Processing. International Journal of Advanced Research in Science, Communication and Technology, 3 (2), 225–227. https://doi.org/10.48175/ijarsct-8687
5. Ramadan, R., Aly, S. (2022). CU-Net: A New Improved Multi-Input Color U-Net Model for Skin Lesion Semantic Segmentation. IEEE Access, 10, 15539–15564. https://doi.org/10.1109/access.2022.3148402
6. Abdelhafeez, A., Mohamed, H. K., Maher, A., Khalil, N. A. (2023). A novel approach toward skin cancer classification through fused deep features and neutrosophic environment. Frontiers in Public Health, 11. https://doi.org/10.3389/fpubh.2023.1123581
7. Salma, W., Eltrass, A. S. (2022). Automated deep learning approach for classification of malignant melanoma and benign skin lesions. Multimedia Tools and Applications, 81 (22), 32643–32660. https://doi.org/10.1007/s11042-022-13081-x
8. Gound, R. S., Gadre, P. S., Gaikwad, J. B., Wagh, P. K. (2018). Skin Disease Diagnosis System using Image Processing and Data Mining. International Journal of Computer Applications, 179 (16), 38–40. https://doi.org/10.5120/ijca2018916253
9. Rao, T. K., Chamanthi, P., Kumar, N. T., Amulya, R. L., Sagar, M. U. (2023). Skin Disease Detection Using Machine Learning. International Journal of Food and Nutritional Sciences, 11 (12). https://doi.org/10.48047/ijfans/v11/i12/171
10. Nguyen, V. D., Bui, N. D., Do, H. K. (2022). Skin Lesion Classification on Imbalanced Data Using Deep Learning with Soft Attention. Sensors, 22 (19), 7530. https://doi.org/10.3390/s22197530
11. Alsahafi, Y. S., Kassem, M. A., Hosny, K. M. (2023). Skin-Net: a novel deep residual network for skin lesions classification using multilevel feature extraction and cross-channel correlation with detection of outlier. Journal of Big Data, 10 (1). https://doi.org/10.1186/s40537-023-00769-6
12. Hosseinzadeh, M., Hussain, D., Zeki Mahmood, F. M., Alenizi, F., Varzeghani, A. N., Asghari, P. et al. (2024). A model for skin cancer using combination of ensemble learning and deep learning. PLOS ONE, 19 (5), e0301275. https://doi.org/10.1371/journal.pone.0301275

13. Sharma, A. K., Tiwari, S., Aggarwal, G., Goenka, N., Kumar, A., Chakrabarti, P. et al. (2022). Dermatologist-Level Classification of Skin Cancer Using Cascaded Ensembling of Convolutional Neural Network and Handcrafted Features Based Deep Neural Network. IEEE Access, 10, 17920–17932. https://doi.org/10.1109/access.2022.3149824
14. Zhang, X., Wang, S., Liu, J., Tao, C. (2018). Towards improving diagnosis of skin diseases by combining deep neural network and human knowledge. BMC Medical Informatics and Decision Making, 18 (S2). https://doi.org/10.1186/s12911-018-0631-9
15. Wan, L., Ai, Z., Chen, J., Jiang, Q., Chen, H., Li, Q. et al. (2022). Detection algorithm for pigmented skin disease based on classifier-level and feature-level fusion. Frontiers in Public Health, 10. https://doi.org/10.3389/fpubh.2022.1034772
16. Foahom Gouabou, A. C., Collenne, J., Monnier, J., Iguernaissi, R., Damoiseaux, J.-L., Moudafi, A., Merad, D. (2022). Computer Aided Diagnosis of Melanoma Using Deep Neural Networks and Game Theory: Application on Dermoscopic Images of Skin Lesions. International Journal of Molecular Sciences, 23 (22), 13838. https://doi.org/10.3390/ijms232213838
17. Tahir, M., Naeem, A., Malik, H., Tanveer, J., Naqvi, R. A., Lee, S.-W. (2023). DSCC_Net: Multi-Classification Deep Learning Models for Diagnosing of Skin Cancer Using Dermoscopic Images. Cancers, 15 (7), 2179. https://doi.org/10.3390/cancers15072179
18. Efimenko, M., Ignatev, A., Koshechkin, K. (2020). Review of medical image recognition technologies to detect melanomas using neural networks. BMC Bioinformatics, 21 (S11). https://doi.org/10.1186/s12859-020-03615-1
19. Lyakhov, P. A., Lyakhova, U. A., Nagornov, N. N. (2022). System for the Recognizing of Pigmented Skin Lesions with Fusion and Analysis of Heterogeneous Data Based on a Multimodal Neural Network. Cancers, 14 (7), 1819. https://doi.org/10.3390/cancers14071819
20. Han, S. S., Moon, I. J., Lim, W., Suh, I. S., Lee, S. Y., Na, J.-I. et al. (2020). Keratinocytic Skin Cancer Detection on the Face Using Region-Based Convolutional Neural Network. JAMA Dermatology, 156 (1), 29. https://doi.org/10.1001/jamadermatol.2019.3807
21. Rasel, M. A., Obaidellah, U. H., Kareem, S. A. (2022). Convolutional Neural Network-Based Skin Lesion Classification With Variable Nonlinear Activation Functions. IEEE Access, 10, 83398–83414. https://doi.org/10.1109/access.2022.3196911
22. Ding, J., Song, J., Li, J., Tang, J., Guo, F. (2022). Two-Stage Deep Neural Network via Ensemble Learning for Melanoma Classification. Frontiers in Bioengineering and Biotechnology, 9. https://doi.org/10.3389/fbioe.2021.758495
23. Hoefler, T., Alistarh, D., Ben-Nun, T., Dryden, N., Peste, A. (2021). Sparsity in deep learning: Pruning and growth for efficient inference and training in neural networks. Journal of Machine Learning Research, 22. Available at: https://jmlr.org/papers/volume22/21-0366/21-0366.pdf
24. Saleem Abdullah, S. M., Abdulazeez, A. M. (2021). Facial Expression Recognition Based on Deep Learning Convolution Neural Network: A Review. Journal of Soft Computing and Data Mining, 02 (01). https://doi.org/10.30880/jscdm.2021.02.01.006
25. Bhanumathi, M., Dhanya, S., Gugan, R., Kirthika, K. G. (2022). Marine Plastic Detection Using Deep Learning. Advances in Parallel Computing Algorithms, Tools and Paradigms, 406–413. https://doi.org/10.3233/apc220057
26. Harumy, T., Ginting, D. S. Br. (2021). Neural Network Enhancement Forecast of Dengue Fever Outbreaks in Coastal Region. Journal of Physics: Conference Series, 1898 (1), 012027. https://doi.org/10.1088/1742-6596/1898/1/012027
27. Ginting, D. S. B., Harumy, T. H. F., Nasution, F. A. F., Siregar, M. E. S. (2021). Analysis Skin Health Patterns in Highlands Area with Apriori and Bayes Contributions. 2021 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA), 83–87. https://doi.org/10.1109/databia53375.2021.9650158
28. Harumy, T. H. F., Chan, H. Y., Sodhy, G. C. (2020). Prediction for Dengue Fever in Indonesia Using Neural Network and Regression Method. Journal of Physics: Conference Series, 1566 (1), 012019. https://doi.org/10.1088/1742-6596/1566/1/012019
29. Harumy, T. H. F., Yustika Manik, F., Altaha (2021). Optimization Classification of Diseases Which is Dominant Suffered by Coastal Areas Using Neural Network. 2021 International Conference on Data Science, Artificial Intelligence, and Business Analytics (DATABIA), 136–141. https://doi.org/10.1109/databia53375.2021.9650223