Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 9 No. 6 (2024) 1385 - 1397 e-ISSN: 2580-0760

CNN-Based Skin Cancer Classification with Combined Max and Global Average Pooling

Chairani Fauzi[1], Fitra Salam S. Nagalay[2]*
[1,2]Institute Informatics and Business Darmajaya, Lampung, Indonesia
[1]chairani@darmajaya.ac.id, [2]fitra.2321211010@mail.darmajaya.ac.id

Abstract

Skin cancer represents a significant threat to human health, with a rising incidence of new cases annually. Timely identification is essential for enhancing recovery rates, however, conventional diagnostic methods such as biopsy are often invasive, time-consuming, and costly. To address this issue, artificial intelligence-based diagnostic systems, particularly Convolutional Neural Networks (CNNs), offer a promising solution for enhancing diagnostic accuracy and efficiency. This study seeks to assess the efficacy of a CNN model that integrates Max Pooling and Global Average Pooling for the detection of skin cancer in digital dermoscopic pictures. The ISIC dataset was used, focusing on two classes, malignant and benign. The combination of Max Pooling and GAP is intended to increase model precision while reducing the ri sk of overfitting. The experimental results show that the proposed model achieved a precision of 96.35%, indicating strong performance in minimizing false positives. However, the recall was relatively low at 85.99%, suggesting reduced sensitivity in detecting malignant cases. The overall accuracy of the combined model was 91.68%, slightly lower than the Max Pooling-only model (91.79%). Although the combination does not significantly improve accuracy, it effectively enhances precision to 96.35%. This is a critical advantage in a clinical setting, as it directly translates to minimizing false positive diagnoses and preventing patients from undergoing unnecessary invasive procedures like biopsies.

Keywords: convolutional neural network; global average pooling; max pooling; pooling; skin cancer

How to Cite: C. Fauzi and F. S. S. Nagalay, "CNN-Based Skin Cancer Classification with Combined Max and Global Average Pooling", *J. RESTI (Rekayasa Sist. Teknol. Inf.)*, vol. 9, no. 6, pp. 1385 - 1397, Dec. 2025.
Permalink/DOI: https://doi.org/10.29207/resti.v9i6.6617

Received: May 2, 2025
Accepted: July 23, 2025
Available Online: December 16, 2025

1. Introduction

The human body's most extensive organ is the skin, functioning as a critical shield against environmental hazards like ultraviolet rays, harmful microorganisms, physical trauma, and toxic substances [1]. Ideally, the skin should be prioritized to remain healthy and disease-free. However, due to poor hygiene, environmental factors, extreme weather conditions, and allergies, the skin becomes vulnerable to various diseases, one of which is skin cancer [2].

Skin cancer is one of the most dangerous diseases globally, with increasing incidence over the past few decades [3]. Data from the World Health Organization (WHO) indicates that over 1.5 million individuals were diagnosed with skin cancer globally in 2022, resulting in approximately 60,000 fatalities [4]. Although the number of cases in Indonesia remains relatively low, early detection is crucial to prevent severe complications or even death if left untreated in later stages [5].

There are two general types of skin cancer, benign and malignant [6]. Malignant malignancies have greater aggressiveness and the ability to metastasize, encompassing disorders such as melanoma, vascular lesions, basal cell, actinic keratosis, and squamous cell. Meanwhile, benign types such as melanocytic nevus and benign keratosis are less dangerous but still require accurate diagnosis [3]. Early diagnosis significantly improves the chance of recovery, with some studies reporting up to 95% survival if detected early [7].

Early detection is often challenging due to the subtlety of initial symptoms and the reliance on the expertise of dermatologists. Inexperienced practitioners may misdiagnose or overlook critical signs [8]. Biopsy, the most commonly used method, involves removing tissue samples and is often painful, slow, and expensive [9].

Therefore, there is a pressing need for faster, less invasive, and more cost-effective diagnostic methods.

The rapid progression of artificial intelligence and machine learning has facilitated the creation of computer-aided diagnostic (CAD) programs that can analyze medical imaging with considerable precision [10]. A multitude of research have successfully utilized deep learning, particularly Convolutional Neural Networks (CNNs), for the classification of medical images [7]. CNNs can automatically extract features from raw images, outperforming traditional diagnostic methods and even rivaling dermatologists in accuracy [11]. However, a significant challenge in CNNs lies in the selection of the appropriate pooling method. A fundamental element of CNN architecture is the pooling layer, a mechanism designed to lower image dimensionality while preserving essential feature information [12]. with two commonly used methods in CNN namely local pooling such as max pooling and global pooling such as global average pooling (GAP) [13]. Max pooling retains the largest value in each pooling area, helping to capture salient local features [14]. On the other hand, GAP pools the average of all features and improves model generalization [15]. Previous studies have often used these pooling techniques independently. For instance, Luqman Hakim applied Max and Average Pooling, achieving only 75% accuracy [7]. Reynaldi Saputro achieved 92.64% accuracy using Max Pooling with different training configurations [16]. Teresia R. Savera compared CNN and K-Nearest Neighbor (KNN) for skin cancer classification, where CNN performed better, though still with modest accuracy [5]. Another study by Lokesh Kumar used GAP in brain tumor classification and achieved a high accuracy of 97.48%, highlighting GAP's potential to reduce overfitting and improve generalization [15]. The data indicate that the integration of Max Pooling and GAP may enhance the advantages of both methods, resulting in improved efficacy in skin cancer categorization. This study aims to develop a CNN model that incorporates both Max Pooling and GAP to detect skin cancer automatically from digital images. Using the International Skin Imaging Collaboration (ISIC) dataset, this research investigates the effect of combining these pooling techniques on the performance of CNN models. It is expected that this combination can enhance detection accuracy while reducing overfitting. This research seeks to augment the reliability and efficiency of skin cancer diagnostic techniques, enabling early detection and enhancing patient outcomes. While many novel CNN architectures have been proposed, there is limited research that specifically investigates the critical trade-off between general classification accuracy and clinical precision. The work seeks to address that deficiency by offering a comprehensive comparative analysis to ascertain how the integration of existing pooling approaches might be maximized for enhanced diagnostic outcomes, with primary focus on decreasing false positives.

2. Research Methods

This research involved several stages to develop and assess a model integrating Max Pooling and Global Average Pooling (GAP) within a Convolutional Neural Network for skin prediction, including literature review, data collection, preprocessing, model construction, training, and evaluation. A detailed description of each stage of this research is presented in Figure 1.

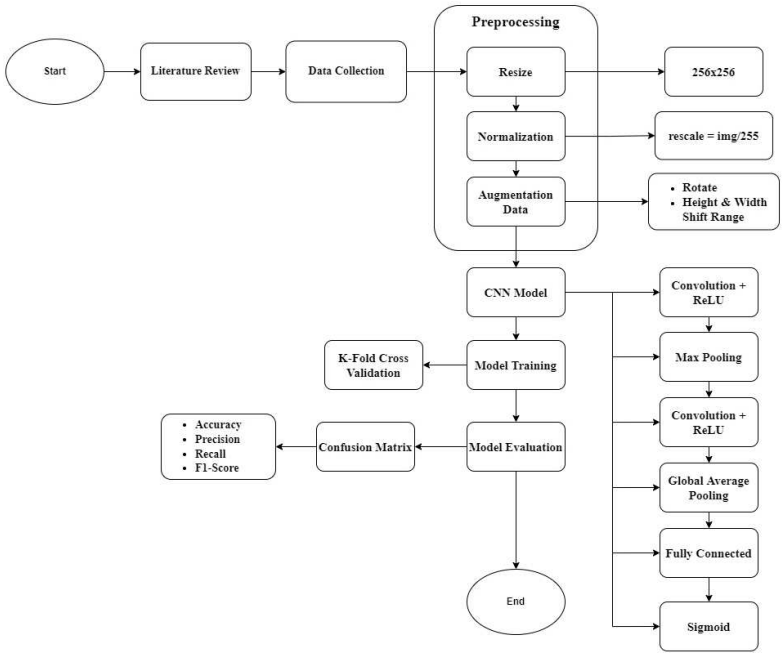

Figure 1. Research Flow for CNN-Based Skin Cancer Classification

The research flow began with a literature review to explore existing approaches in medical image classification, particularly those involving deep learning and pooling strategies in CNN architectures. This step also helped identify gaps in previous studies, where most have implemented either Max Pooling or GAP in isolation, without exploring the combined effect of both.

2.1 Data Collection

At this stage, a dataset is obtained from the ISIC Archive dataset (International Skin Imaging Collaboration), which provides a large set of labeled dermoscopic images for both benign and malignant skin cancer [17]. 5,500 benign and 5,105 malignant images were used, resulting in a slight imbalance between the two classes. To mitigate the potential impact of this imbalance, class weighting was applied during model training, ensuring that the minority class was given appropriate emphasis [18]. Furthermore, data augmentation was implemented during the preprocessing stage to enrich image diversity and improve the model generalization capability [19]. Samples of benign and malignant image data are shown in Figure 2.

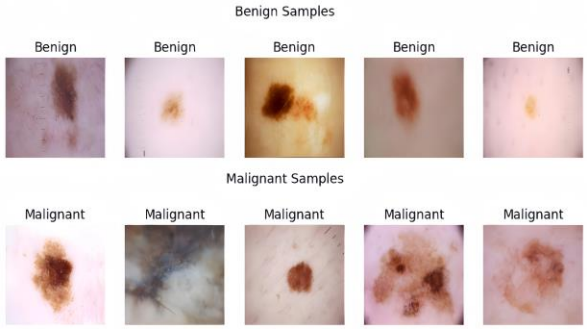

Figure 2. Benign and Malignant Samples

2.2 Preprocessing Data

Following data collection, the study implements a preprocessing framework consisting of image resizing, normalization, and augmentation. These steps are requisite to ensure scalar uniformity and to bolster the model's robustness when encountering unseen data [19]. To address the inherent dimensional variations within the ISIC dataset, all images were standardized to a uniform resolution of 256x256 pixels. The selection of this size was based on an experiment conducted by Suhendro Y. Irianto et al. [20], which demonstrated that although the difference in validation accuracy between various image sizes was not significant, the execution time remained within an acceptable range. The experiment showed that resizing images to 256×256 pixels resulted in a training accuracy of 0.9721 and a validation accuracy of 0.9520, with an average execution time of 26.4 seconds. These results indicate that this resolution offers an ideal trade-off between computational efficiency and classification accuracy.

Following resizing, normalization was performed to adjust the pixel value scale to a consistent range between 0 and 1 [21]. The raw .jpg images consist of RGB channels (Red, Green, Blue), where pixel intensities vary between 0 and 255 [22]. This wide range can negatively impact model training, as neural networks tend to converge faster when input values are within a smaller, standardized range [23]. Therefore, in this step, each pixel value in the image was divided by 255 to produce a normalized matrix, as represented in Equation 1.

$$Rescale = img/255 \quad (1)$$

img represents the pixel matrix of the image. This process ensures that each pixel value lies within a consistent scale, enhancing training efficiency and contributing to a more stable learning process [24].

The final preprocessing step was data augmentation, applied to enrich the dataset by introducing variations to existing images. Several augmentation techniques were implemented, including a rotation range of 10 degrees, as well as width and height shift ranges of 0.1. This allows images to be rotated and shifted up to 10% in both vertical and horizontal directions. To handle the empty areas generated by these transformations, the fill mode was set to "nearest", which replaces missing pixels with the value of the nearest neighboring pixel [25]. These strategies are extensively employed to augment the training dataset and facilitate the model's acquisition of more generalized patterns, rather than depending exclusively on the original visual patterns. As a result, augmentation is essential for enhancing model resilience and reducing overfitting during training [15].

2.3 CNN Model

Following the completion of data preprocessing phase, the study proceeded to the construction of the CNN architecture. CNN are extensively utilized in digital image analysis, distinguished by their superior accuracy and robust capacity for identifying intricate patterns [26]. Here's the proposed CNN model, shown in Figure 3.

The CNN architecture created in this study is specifically designed for the binary classification of skin lesions. As depicted in Figure 3, the network accepts input images with dimensions of 256×256 pixels. Structurally, the model is composed of two primary distinct segments, the feature extraction layers and the classification layers [27].

The feature extraction component consists of three convolutional blocks. Each block includes two Conv2D layers followed by a Rectified Linear Unit (ReLU) activation function and batch normalization, which helps stabilize the training process and improve convergence speed [28], [29]. Max pooling is implemented following each block to downsample

spatial dimensions and minimize computational complexity, while effectively retaining the most salient features [30]. To further improve generalization and prevent overfitting, Dropout is introduced after each block [31].

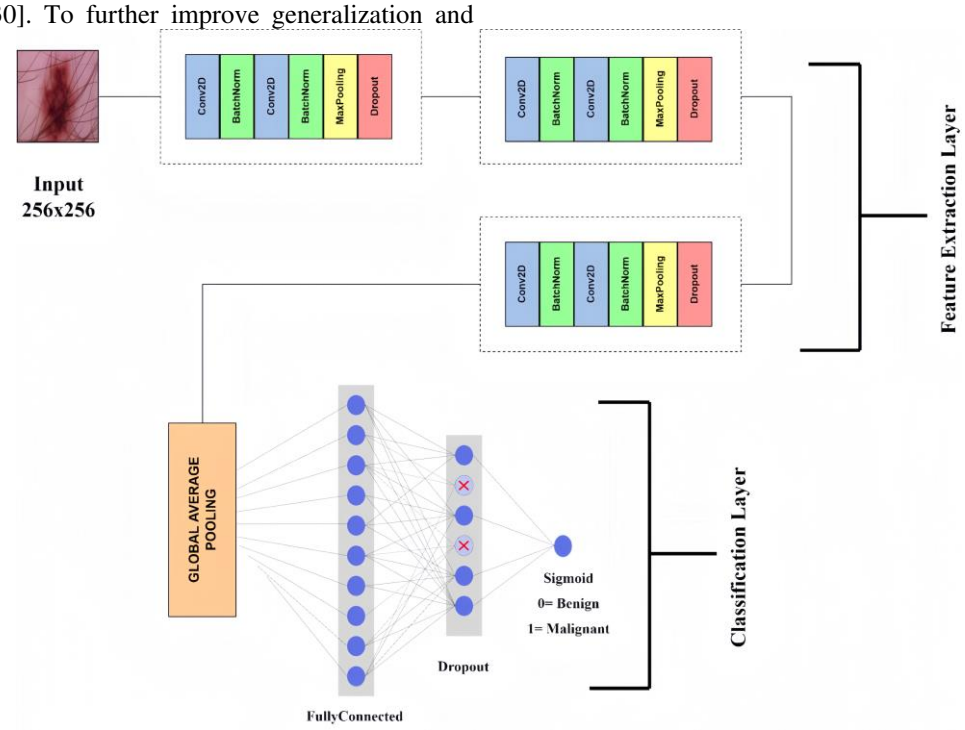

Figure 3. The Proposed CNN Architecture Design

As the input data progresses through the layers, the number of filters increases progressively. This hierarchical architecture facilitates the acquisition of progressively abstract and complex feature representations. Initial layers detect elementary patterns, such as edges and textures, whereas deeper layers discern more intricate structures, including lesion morphology and boundaries. Consequently, this multi-level learning capability renders CNNs highly effective for classification tasks, as it eliminates the necessity for manual feature engineering [32].

Following the feature extraction phase, the output is processed by a Global Average Pooling (GAP) layer. GAP downsamples the spatial dimensions by calculating the mean value of each feature map. This operation streamlines data representation and minimizes the volume of trainable parameters, effectively serving as a mechanism to mitigate overfitting [33].

The output generated by the GAP layer is subsequently propagated to a Fully Connected layer, followed by an additional Dropout layer to reinforce network regularization. The architecture culminates in a sigmoid activation function within the output layer, producing a probability value between 0 and 1 for the final categorization [34], 0 for benign and 1 for malignant.

2.4 Training Model

The model was trained using k-fold cross validation, a commonly used statistical method. K-fold cross-validation mitigates undue reliance on a particular data subset and offers a more reliable and stable assessment of the model's generalization capability by utilizing the entire dataset for both training and testing [35].

The dataset is partitioned into k subsets of comparable size, commonly referred to as folds. During each cycle, the model utilizes k-1 folds for training purposes, while the single remaining fold serves as the validation data. This cycle repeats until every specific fold has functioned as the validation set exactly once. After all iterations, the performance metrics like as loss and accuracy are averaged to get a final estimate of the model's performance. This method assures that no data is wasted and that the evaluation takes into consideration the variety in data distribution.

2.5. Evaluation Model

To validate the model trained with k-fold cross-validation, a confusion matrix was employed to quantify performance deviations between predicted and actual classes. This approach provides a comprehensive assessment of the categorization capabilities by breaking down the output into four major metrics, True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) [35]. True Positives denote the quantity of positive instances accurately classified by the model, whereas False Positives represent negative instances erroneously categorized as positive. Conversely, False Negatives refer to positive cases incorrectly labeled as negative, whereas True Negatives

measure the accurately detected negative instances. Based on these values, several evaluation metrics are derived to assess the classification performance. Accuracy, as shown in Equation 2, measures the overall correctness of the model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

Precision, defined in Equation 3, measures the ratio of accurately identified positive predictions to the total number of positive predictions.

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

Recall, presented in Equation 4, denotes the model's capacity to identify true positive cases (sensitivity).

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

F1-Score in Equation 5 represents the harmonic mean of precision and recall, providing a balance between the two metrics

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

These metrics provide detailed insight into the model's strengths and weaknesses in distinguishing between the two classes [36]. For critical tasks like skin cancer screening, Recall is of particular significance, as minimizing false negatives is essential to prevent the adverse consequences of missed diagnoses [37], [38]. Furthermore, the Area Under the Receiver Operating Characteristic Curve was computed to assess the overall effectiveness of each model in differentiating between benign and malignant categories [39].

3. Results and Discussions

Following the research stages outlined previously, this study has been fully executed, ranging from data acquisition to performance assessment. The process began by gathering dermoscopic images from the ISIC Archive, which were utilized as the primary dataset to train and validate the CNN model. A total of 10,605 images were used in this study, consisting of 5,500 images labeled as benign and 5,105 images labeled as malignant. This balanced distribution ensured that the model could learn effectively from both classes and avoid bias toward one category.

After collecting the data, all images were resized to a consistent dimension of 256×256 pixels to standardize input dimensions and facilitate efficient training. The next step was image normalization, in which pixel values originally ranging from 0 to 255 were rescaled to a range between 0 and 1. The purpose of this normalization was to ensure uniform data scaling, allowing the neural network to train more stably and converge more quickly.

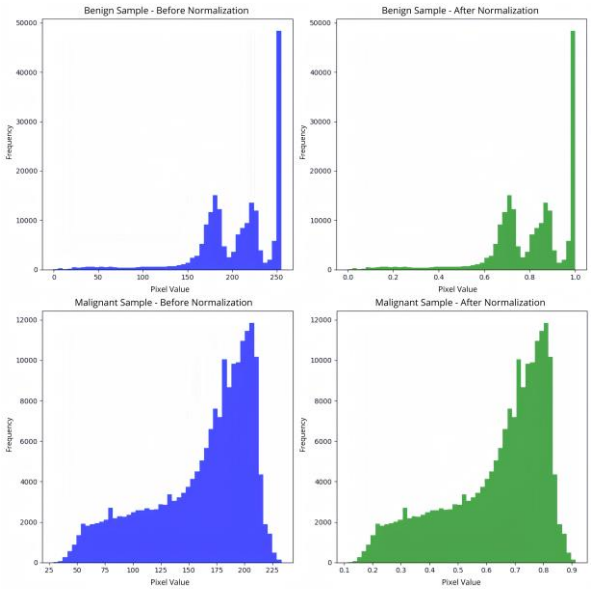

Figure 4. The Pixel Distribution of RGB Channels Benign and Malignant Before and After Normalization

Figure 4 illustrates the distribution of pixel values in both benign and malignant image sets before and after normalization. The graphs show that the overall distribution patterns remain similar, but the values are compressed into a smaller scale.

Following normalization, data augmentation was applied to enrich the dataset with varied image samples and improve the model's ability to generalize [40]. Augmentation techniques used in this study included rotating the images and shifting them horizontally and vertically by 10% of the image size. These transformations were applied to increase dataset variability while preserving lesion structure.

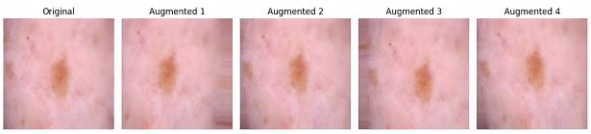

Figure 5. Original and Augmented Image Samples in Benign Class

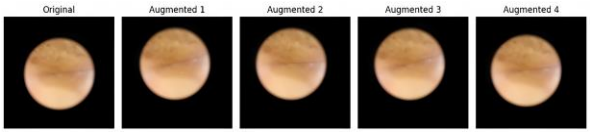

Figure 6. Original and Augmented Image Samples in Malignant Class

Figures 5 and 6 display sample images from the benign and malignant classes, respectively, before and after augmentation. The augmented samples show realistic variations in orientation and position, helping to improve the model to become more robust in detecting lesions under different conditions [41], [42].

After the preprocessing process is completed, this study proceeded with the implementation of two CNN model architectures for analysis. The first architecture is the

proposed model, which combines Max Pooling and Global Average Pooling, as summarized in Table 1.

Table 1. Model summary of the proposed CNN architecture using a combination of Max Pooling and Global Average Pooling

Layer Block	Layer Type	Filters/ Units	Kernel Size	Activation
Block 1	Conv2D, Conv2D	32	(3,3)	ReLU
	Batch Normalization	-	-	-
	MaxPooling2D	-	(2,2)	-
	Dropout	0.2	-	-
Block 2	Conv2D, Conv2D	64	(3,3)	ReLU
	Batch Normalization	-	-	-
	MaxPooling2D	-	(2,2)	-
	Dropout	0.3	-	-
Block 3	Conv2D, Conv2D	128	(3,3)	ReLU
	Batch Normalization	-	-	-
	MaxPooling2D	-	(2,2)	-
	Dropout	0.2	-	-
Classifier	GlobalAveragePooling2D	128	-	-
	Dense	256	-	ReLU
	Dropout	0.5	-	-
	Dense (Output)	1	-	Sigmoid

The proposed architecture is organized into sequential convolutional blocks. The initial block comprises two Convolutional (Conv2D) layers equipped with 32 filters. Then, a second convolutional layer with the same configuration is applied, followed by batch normalization again to further improve stability [43]. After these two convolution layers, a max pooling layer is used to reduce the spatial dimensionality of the extracted features [44], as well as a dropout layer 0.2 to reduce overfitting [45]. Furthermore, the second block expands the network depth to two convolutional layers with 64 filters, each followed by a batch normalization layer to stabilize the activation. After that, a max pooling layer with a pool size of 2x2 is again applied, Subsequently, a dropout layer with a rate of 0.3 is integrated to mitigate the risk of overfitting. The third block features a pair of convolutional layers equipped with 128 filters and a 3x3 kernel size, both utilizing ReLU activation and succeeded by a Batch Normalization layer. After these two convolution layers, a max pooling layer with a pool size of 2x2 is used to further reduce the spatial dimensionality, followed by a dropout layer 0.2.

Once the convolution and pooling process is complete, the model uses a Global Average Pooling (GAP) layer to produce a one-dimensional vector of size 128. This process helps to capture important information from all the extracted features and reduces the spatial dimension significantly. Then, there is a dense layer with 256 units that uses the ReLU activation function and L2 regularization to strengthen the feature representation, followed by a dropout layer with a level of 0.5. Finally, the model has an output layer with 1 unit that uses a sigmoid activation [46].

To compare performance, the second model architecture is built using only Max Pooling, without the GAP component. The structure follows a similar convolutional block pattern as the proposed model, as detailed in Table 2.

In this model summary max pooling alone without combination starts with the first convolution layer (Conv2D) which has 32 filters with a 3x3 kernel size, using the ReLU activation function. This layer is followed by batch normalization to stabilize the activation distribution during training. The second convolution layer also has 32 filters and a 3x3 kernel size, followed by batch normalization. After that, a max pooling operation (MaxPooling2D) using a 2×2 pool size to downsample the spatial features. Moving to the second block, the network implements a convolutional layer containing 64 filters with 3×3 kernels, immediately followed by batch normalization for improved training stability. This is succeeded by another identical convolutional configuration (64 filters, 3×3 kernels) with batch normalization. The block concludes with another 2×2 max pooling operation for further spatial reduction. In the third block, the network doubles the filter count with a convolutional layer using 128 filters (3×3 kernels) paired with batch normalization. This is followed by an identical convolutional setup (128 filters, 3×3 kernels) also with batch normalization. The block finishes with a final 2×2 max pooling layer to additionally compress the spatial dimensions.

Upon completion of the convolution and pooling stages, the feature maps are reshaped into a one-dimensional vector. The resulting vector is then fed into a dense layer comprising 256 units, utilizing the ReLU activation function alongside L2 regularization to

improve the robustness of the extracted features. Subsequently, a dropout rate of 0.5 is implemented to mitigate overfitting. The architecture concludes with an output layer utilizing a sigmoid activation function to execute binary classification between malignant and benign cases.

Table 2. Model summary of the baseline CNN architecture using Max Pooling only.

Layer Block	Layer Type	Filters/ Units	Kernel Size	Activation
Block 1	Conv2D, Conv2D	32	(3,3)	ReLU
	Batch Normalization	-	-	-
	MaxPooling2D	-	(2,2)	-
	Dropout	0.2	-	-
Block 2	Conv2D, Conv2D	64	(3,3)	ReLU
	Batch Normalization	-	-	-
	MaxPooling2D	-	(2,2)	-
	Dropout	0.3	-	-
Block 3	Conv2D, Conv2D	128	(3,3)	ReLU
	Batch Normalization	-	-	-
	MaxPooling2D	-	(2,2)	-
	Dropout	0.2	-	-
Classifier	Flatten	-	-	-
	Dense	256	-	ReLU
	Dropout	0.5	-	-
	Dense (Output)	1	-	Sigmoid

Both CNN models were trained and tested using k-fold cross-validation. In this study, the parameter k was established at 5, signifying that the dataset was partitioned into five equal-sized folds. During each iteration, four folds were utilized for training, while the remaining fold functioned as validation data. The dataset was split proportionally to preserve the original distribution between benign and malignant classes across all folds, ensuring that each subset reflected a balanced representation.

Each fold was trained for 50 epochs using preprocessed images from both classes. The training aimed to evaluate the consistency and stability of the models' performance over multiple subsets of the data. To address the slight class imbalance, class weighting was applied during training by calculating weights based on the frequency of each class in the training data for every fold. This approach helped reduce model bias toward the majority class (benign) and improved sensitivity to the minority class (malignant). The subsequent figure depicts the accuracy and loss curves for the model employing the combination model over folds 1, 3, and 5 in Figure 7.

Based on the accuracy and loss graphs for each fold, several observations can be made. In Fold 1, training accuracy shows a consistent upward trend from approximately 0.80 to 0.95. Validation accuracy fluctuates in the early epochs but stabilizes around 0.85 toward the end of training. Training loss steadily decreases from 0.8 to 0.2, while validation loss is more erratic but ultimately stabilizes near 0.4. For Fold 3, training accuracy increases from 0.85 to 0.90. Validation accuracy begins at 0.60 and stabilizes around 0.85 by the end of training. Training loss drops from 0.5 to 0.25, while validation loss shows fluctuations with some spikes but eventually settles around 0.5. Finally, in Fold 5, training accuracy rises from 0.75 to over 0.90. Validation accuracy exhibits significant fluctuation, beginning at 0.50 and peaking above 0.90, indicating substantial instability. Training loss decreases steadily, while validation loss presents sharp spikes, suggesting periods of instability.

This shows for the training performance of the model its good ability to capture patterns from the training data with increasing training accuracy and decreasing training loss consistently across all folds. Then the performance on validation accuracy is generally stable at the end of training although there are fluctuations at the beginning. However, significant fluctuations in validation loss indicate a potential overfitting or instability problem during training [47].

The training and testing performance of the baseline CNN model, which uses only Max Pooling without the combination of GAP, is shown in the following accuracy and loss graphs for folds 1, 3, and 5 in Figure 8.

Based on the graphs, the following observations can be made for each fold. In Fold 1, training accuracy improves steadily from approximately 0.70 to around 0.90 as the number of epochs increases. Validation accuracy follows a similar trend, although it exhibits significant fluctuations early in training. Toward the final epochs, validation accuracy stabilizes around 0.90. The training loss decreases consistently from about 2.0 to near 0, and validation loss shows a similar pattern with slight variations. Regarding Fold 3, training accuracy increases from 0.70 to approximately 0.90,

with validation accuracy following a similar path. Although there are notable fluctuations during the early training stages, validation accuracy eventually stabilizes around 0.90. Training loss decreases steadily from approximately 2.0 to near 0, while validation loss initially fluctuates but stabilizes toward the end of the training process. Lastly, for Fold 5, training accuracy improves consistently from 0.70 to above 0.90. Validation accuracy shows significant fluctuations throughout training but ultimately settles around 0.90. Training loss again shows a steady decline from 2.0 to nearly 0, while validation loss displays irregular variations, indicating instability during certain phases of training.

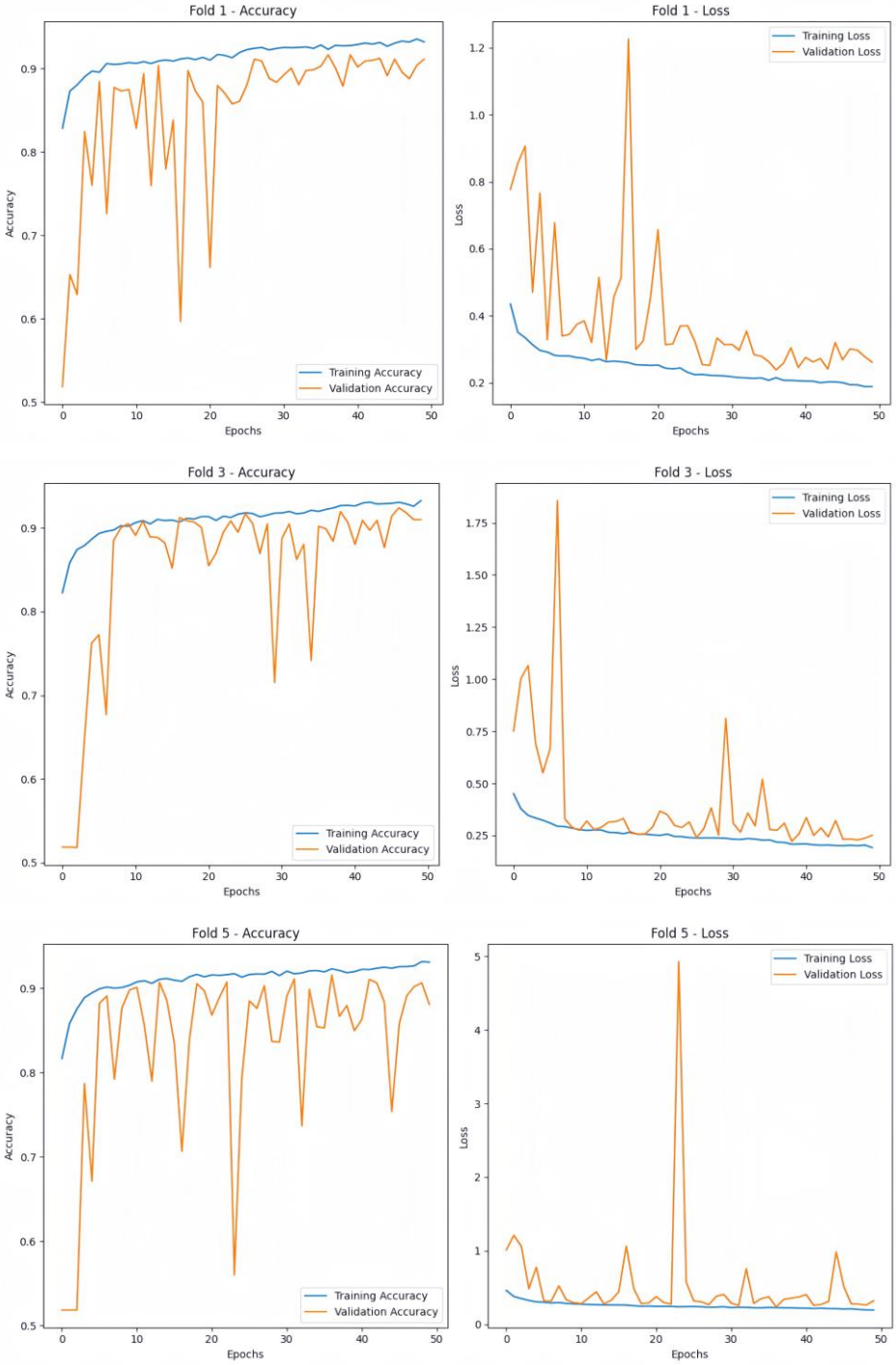

Figure 7. Accuracy and loss graphs of the proposed CNN model (Max Pooling + GAP) for folds 1, 3, and 5.

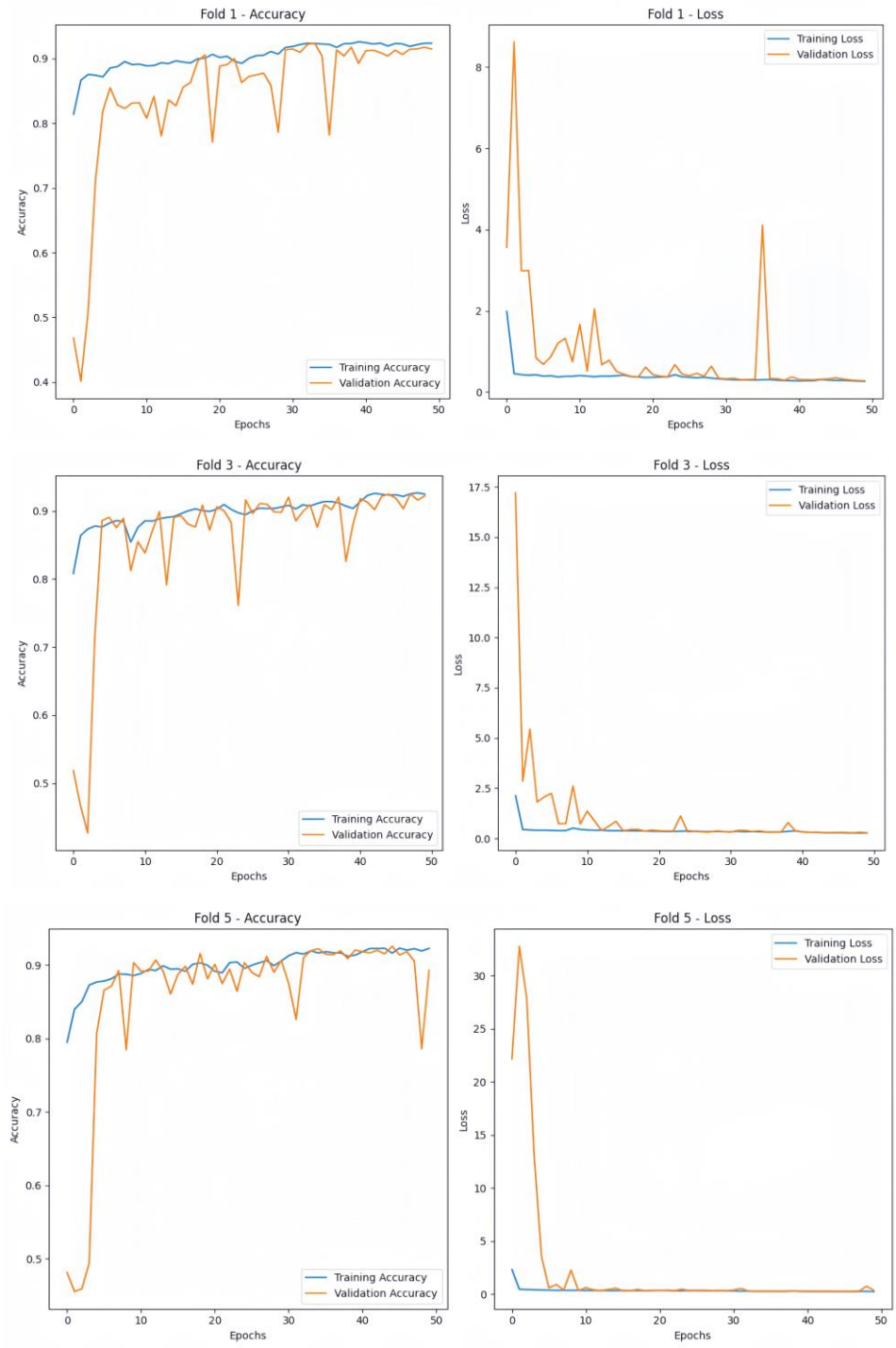

Figure 8. Accuracy and loss graphs of the baseline CNN model (Max Pooling Only) for folds 1, 3, and 5.

The results indicate that the Max Pooling only model is capable of achieving relatively high training and validation accuracy, similar to the proposed model. However, the presence of more pronounced fluctuations in validation loss across all folds may suggest that the model is more susceptible to instability or potential overfitting, particularly when compared to the more stabilized performance of the proposed model.

Based on the graphical results of the training and testing performance process of the two models. The comparison between the max pooling and global average pooling combination model and the max pooling only model without combination is that the combination model has fluctuations in validation loss at the beginning and middle after which it tends to stabilize at the end of training. While the max pooling-only model shows rapid stabilization after high initial fluctuations, with validation accuracy that eventually

approaches training accuracy and loss that remains low after the first few epochs. This is quite good, as after the initial fluctuations, the model quickly stabilizes and maintains good performance. In line with the observation in Prechelt's study [48], the validation error curve often has several minimum points before reaching the best result. The combined model experienced more fluctuations before stabilizing, while the max pooling model alone stabilized faster after a poor training start. Overall, the combination model had a more variable learning pattern but remained stable at the end of training, while the max pooling-only model showed a faster stabilization process after high initial fluctuations. This shows that the max pooling model does not suffer from severe overfitting, but instead achieves a balance between training and generalization. This is due to the use of other regulation techniques such as Dropout or Batch Normalization, then the training data is diverse and large enough that overfitting can be controlled by Max Pooling [49], [50].

Table 3 shows a comparative analysis of the average assessment outcomes between the suggested model (Max Pooling + GAP) and the baseline model (Max Pooling only).

Table 3. Comparison of Average Evaluation Results Between Models.

CNN Model	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	AUC (%)
Max Pooling + GAP	91.68	96.35	85.99	90.87	94.82
Max Pooling Only	91.79	93.60	89.04	91.26	97.36

Based on the evaluation results in Table 3, an interesting trade-off between the two models becomes apparent. The Max Pooling Only model demonstrates superior performance across several general metrics. It achieved a slightly higher accuracy (91.79%), recall (89.04%), F1-Score (91.26%), and a notably higher AUC of 97.36%. This superior AUC value, visualized in Figure 9, indicates that the Max Pooling Only architecture is generally more robust at discriminating between benign and malignant lesions across all thresholds.

However, the primary contribution and most critical advantage of our proposed model (Max Pooling + GAP) lie in its significantly superior precision (96.35% vs. 93.60%). This sharp increase in precision highlights that the combined model is effectively optimized for a more specific and clinically vital purpose: minimizing false positives. In medical applications, where a misdiagnosis can lead to patients undergoing painful, costly, and unnecessary biopsies, the ability of a model to be highly trustworthy when predicting a positive case is an invaluable asset. Therefore, this study demonstrates that while not outperforming on all metrics, the proposed combined model offers a specialized solution that is safer and more reliable for clinical scenarios where precision is the highest priority.

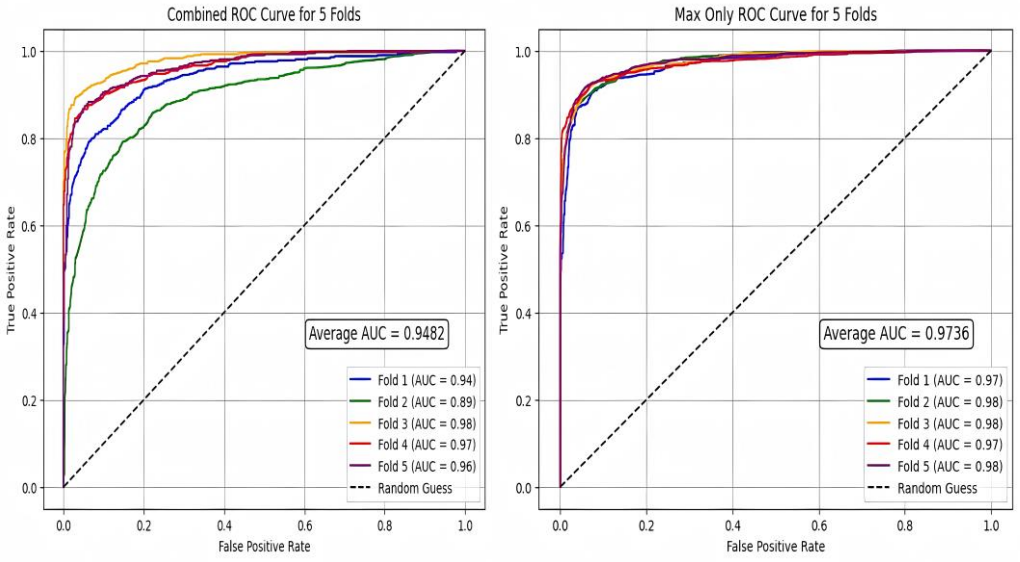

Figure 9. Comparison of ROC Curves for the combined model and the max pooling only

To further assess the model's behavior, confusion matrices were generated for each fold. These matrices visualize the number of correct and incorrect predictions for both benign and malignant categories, and help interpret where the models performed well or made errors. The confusion matrices for the proposed model and the baseline model are presented in Table 4 and Table 5, respectively.

Based on the comparison of the results table of the confusion matrix on both models, there is a significant

difference in the pattern of skin cancer detection performance. In the combination model (Max Pooling + GAP), the true positive value indicates the number of malignant skin cancer cases that were correctly detected by the model. For example, in fold 1, the number 927 indicates that 927 malignant cancer samples were correctly identified. This TP value tends to vary across folds, with fold 1 having a lower TP value than the other folds, indicating different detection consistency depending on the distribution of data within the fold.

The combined model showed superiority in terms of True Negative (TN), such as in fold 1 with a value of 1016, meaning 1016 benign skin cases were correctly categorized as benign. A significant advantage of the combination model is seen in the consistently lower False Positive (FP) values across folds. For example, in fold 2, the combined model only produced 27 FP cases (benign cases that were misclassified as malignant), while the Max Pooling model alone produced 53 FP cases. This lower FP value indicates better precision, in line with the higher precision value in the evaluation table (96.35% vs 93.60%).

However, the trade-off is evident in the False Negative (FN) values, where the combined model has higher numbers in most folds. In fold 2, there were 169 FN cases (undetected malignant cases) compared to only 104 cases in the Max Pooling model alone. This higher FN value indicates that the combined model tends to be more conservative in classifying samples as malignant cancer, which correlates with a lower recall value (85.99% vs 89.04%).

Table 4. Confusion matrix for max pooling + GAP model.

Fold	TP	TN	FP	FN
1	927	1016	84	94
2	1073	852	27	169
3	1070	890	30	131
4	1075	874	25	147
5	1075	880	25	141

Table 5. Confusion matrix for max pooling only model.

Fold	TP	TN	FP	FN
1	1034	906	66	115
2	1047	917	53	104
3	1053	909	47	112
4	1027	895	73	126
5	1028	919	72	102

In the Max Pooling model alone, a different pattern is seen with more consistent and generally higher TP values across all folds. For example, in fold 1, this model correctly identified 1034 malignant cases, compared to 927 in the combined model. TN values are also quite stable, as in fold 3 with 909 correctly identified benign cases. However, the main drawback of this model is the higher FP value, with an example in fold 4 where 73 benign cases were incorrectly categorized as malignant, which could have implications for unnecessary levels of medical intervention in a clinical context.

These differences explain why the Max Pooling model has better recall due to its ability to detect more positive cases, but at the cost of lower precision due to higher FP. The slightly higher F1-Score value for the Max Pooling model alone (91.26% vs 90.87%) indicates that this model achieves a better balance between precision and recall, although the difference is not significant.

Performance differences among the folds reflect the model's sensitivity to specific data distributions. This provides a comprehensive assessment of robustness across various data segments, demonstrating the primary advantage of utilizing k-fold cross-validation for evaluation.

4. Conclusions

Based on the results, several conclusions can be drawn regarding the research objectives that were initially defined. The proposed CNN model that combines Max Pooling and Global Average Pooling (GAP) demonstrated solid performance in skin cancer prediction, achieving an overall accuracy of 91.68%. This level of accuracy indicates the model's strong capability in correctly classifying the majority of skin lesion cases, whether benign or malignant. These findings confirm that a CNN architecture incorporating both pooling techniques is effective and applicable for digital image-based skin cancer detection systems.

One of the most notable strengths of the proposed model is its precision, which reached 96.35%, outperforming the baseline model. This result highlights the model's ability to significantly reduce false-positive cases where benign lesions are incorrectly classified as malignant. Such precision is highly valuable in clinical settings, as it can help avoid overdiagnosis and unnecessary patient anxiety.

While the proposed model is slightly outperformed on several general metrics such as accuracy and AUC, this is a justified trade-off for the substantial and clinically vital gain in precision. This study demonstrates that incorporating Global Average Pooling (GAP) is not merely about boosting a single accuracy metric, but about optimizing the model for diagnostic reliability and safety. Therefore, it can be concluded that the combined pooling approach offers an intelligent and valuable trade-off. While it may not yield the highest absolute accuracy, it contributes more meaningfully by enhancing precision, which is a paramount priority in medical applications where diagnostic errors can have serious consequences.

However, this study has some limitations. The analysis was conducted on a single dataset (ISIC), and the dataset contained a slight class imbalance which,

despite the use of class weighting and data augmentation, may not be fully resolved. Furthermore, as our contribution focuses on the analysis of existing components, future work can be directed towards enhancing the originality and impact of the architectural design. Promising avenues for future research include proposing a novel hybrid pooling mechanism or an adaptive selection method to combine the strengths of different pooling strategies dynamically, applying and evaluating the architecture on a multi-class classification problem (e.g., distinguishing melanoma, carcinoma, and nevus) to assess its robustness on more complex tasks, and performing validation on an external dataset to rigorously test the model's generalization capabilities. This study provides a foundational analysis that opens the door for these future investigations.

Acknowledgements

Fitra Salam S. Nagalay gratefully acknowledges the guidance and support of Dr. Chairani, S.Kom., M.Kom., from the Department of Informatics, Institute of Informatics and Business Darmajaya, whose expertise in artificial intelligence contributed significantly to the direction and depth of this research. The author also appreciates the assistance of academic peers and institutional support that made the completion of this study possible. Special thanks are extended to the ISIC Archive for providing the publicly accessible dataset used in this research.

References

[1] M. Duarte, S. S. Pedrosa, P. R. Khusial, and A. R. Madureira, "Exploring the interplay between stress mediators and skin microbiota in shaping age-related hallmarks: A review," *Mech. Ageing Dev.*, vol. 220, no. February, p. 111956, 2024, doi: 10.1016/j.mad.2024.111956.

[2] E. R. Parker, J. Mo, and R. S. Goodman, "The dermatological manifestations of extreme weather events: A comprehensive review of skin disease and vulnerability," *J. Clim. Chang. Heal.*, vol. 8, no. July 2022, p. 100162, 2022, doi: 10.1016/j.joclim.2022.100162.

[3] E. S. Nugroho, I. Ardiyanto, and H. A. Nugroho, "Systematic literature review of dermoscopic pigmented skin lesions classification using convolutional neural network (CNN)," *Int. J. Adv. Intell. Informatics*, vol. 9, no. 3, pp. 363–382, 2023, doi: 10.26555/ijain.v9i3.961.

[4] International Agency for Research on Cancer (IARC), "Skin cancer," *IARC Newsletter*, 2022. https://www.iarc.who.int/cancer-type/skin-cancer (accessed Oct. 29, 2024).

[5] S. Taresia R, W. H. Suryawan, and A. W. Setiawan, "Early detection of skin cancer using K-NN and convolutional neural network," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 7, no. 2, pp. 373–378, 2020, doi: 10.1142/9488.

[6] A. Bassel, A. B. Abdulkareem, Z. A. A. Alyasseri, N. S. Sani, and H. J. Mohammed, "Automatic malignant and benign skin cancer classification using a hybrid deep learning approach," *Diagnostics*, vol. 12, no. 10, 2022, doi: 10.3390/diagnostics12102472.

[7] L. Hakim, Z. Sari, and Handhajani, "Classification of skin cancer pigment images using convolutional neural network," *J. Resti*, vol. 1, no. 10, pp. 379–385, 2021, [Online]. Available: https://jurnal.iaii.or.id/index.php/RESTI/article/view/3001

[8] K. Vayadande, A. A. Bhosle, R. G. Pawar, D. J. Joshi, P. A. Bailke, and O. Lohade, "Innovative approaches for skin disease identification in machine learning: A comprehensive study," *Oral Oncol. Reports*, vol. 10, no. April, p. 100365, 2024, doi: 10.1016/j.oor.2024.100365.

[9] A. F. Mavrogenis, P. Altsitzioglou, S. Tsukamoto, and C. Errani, "Biopsy techniques for musculoskeletal tumors: basic principles and specialized techniques," *Curr. Oncol.*, vol. 31, no. 2, pp. 900–917, 2024, doi: 10.3390/curroncol31020067.

[10] S. Rajarajeswari, J. Prassanna, M. Abdul Quadir, J. Christy Jackson, S. Sharma, and B. Rajesh, "Skin cancer detection using deep learning," *Res. J. Pharm. Technol.*, vol. 15, no. 10, pp. 4519–4525, 2022, doi: 10.52711/0974-360X.2022.00758.

[11] J. Triloka, H. Hartono, and S. Sutedi, "Detection of SQL injection attack using machine learning based on natural language processing," *Int. J. Artif. Intell. Res.*, vol. 6, no. 2, 2022, doi: 10.29099/ijair.v6i2.355.

[12] M. A. Saputra, Reyhan Adi Putra, Davito Rasendriya Rizqullah Asyrofi, "Implementation of convolutional neural network (CNN) to detect mask usage in images," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3, pp. 710–714, 2023, doi: 10.23960/jitet.v11i3.3286.

[13] A. Zafar *et al.*, "A Comparison of Pooling methods for convolutional neural networks," *Appl. Sci.*, vol. 12, no. 17, pp. 1–21, 2022, doi: 10.3390/app12178643.

[14] J. J. Liu, Q. Hou, Z. A. Liu, and M. M. Cheng, "PoolNet+: exploring the potential of pooling for salient object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 1, pp. 887–904, 2023, doi: 10.1109/TPAMI.2021.3140168.

[15] R. L. Kumar, J. Kakarla, B. V. Isunuri, and M. Singh, "Multi-class brain tumor classification using residual network and global average pooling," *Multimed. Tools Appl.*, vol. 80, no. 9, pp. 13429–13438, 2021, doi: 10.1007/s11042-020-10335-4.

[16] R. R. Saputro, A. Junaidi, and W. A. Saputra, "Classification of skin cancer disease using convolutional neural network method (case study: melanoma)," *J. Dinda Data Sci. Inf. Technol. Data Anal.*, vol. 2, no. 1, pp. 52–57, 2022, doi: 10.20895/dinda.v2i1.349.

[17] A. Nawaz, Khadija Zanib, Atika Shabir, Iqra Li, Jianqiang Wang, Yu Mahmood, Tariq Rehman, "Skin cancer detection using dermoscopic images with convolutional neural network," *Sci. Rep.*, pp. 254–258, 2025, doi: 10.1038/s41598-025-91446-6.

[18] M. Saini and S. Susan, "Tackling class imbalance in computer vision: a contemporary review," *Artif. Intell. Rev.*, vol. 56, no. July, pp. 1279–1335, 2023, doi: 10.1007/s10462-023-10557-6.

[19] J. Wang *et al.*, "Generalizing to unseen domains: a survey on domain generalization," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 8, pp. 8052–8072, 2023, doi: 10.1109/TKDE.2022.3178128.

[20] S. Y. Irianto, R. Yunandar, M. S. Hasibuan, D. A. Dewi, and N. Pitsachart, "Early identification of skin cancer using region growing technique and a deep learning algorithm," *HighTech Innov. J.*, vol. 5, no. 3, pp. 640–662, 2024, doi: 10.28991/HIJ-2024-05-03-07.

[21] E. Academy, "Mengapa perlu menormalkan nilai piksel sebelum melatih model?," *EITCA Academy*, 2023. https://id.eitca.org/kecerdasan-buatan/eitc-ai-tf-tensorflow-fundamental/aliran-tensor-js/menggunakan-tensorflow-untuk-mengklasifikasikan-gambar-pakaian/ulasan-pemeriksaan-tensorflow-js-mengapa-perlu-menormalkan-nilai-piksel-sebelum-melatih-model/ (accessed Nov. 21, 2024).

[22] W. Bakx, B. Gorte, and K. Grabmaier, *The core of GIScience: A process-based approach*. ITC, Faculty of geo-information Science and Earth Observation, 2011.

[23] E. Beyazit, J. Kozaczuk, B. Li, V. Wallace, and B. Fadlallah

Amazon, "An inductive bias for tabular deep learning," *Conf. Neural Inf. Process. Syst.*, no. NeurIPS, 2023.
[24] I. Pacal, "A novel Swin transformer approach utilizing residual multi-layer perceptron for diagnosing brain tumors in MRI images," *Int. J. Mach. Learn. Cybern.*, vol. 15, no. 9, pp. 3579–3597, 2024, doi: 10.1007/s13042-024-02110-w.
[25] G. P. H. P. Gusti, E. Haerani, F. Syafria, F. Yanto, and S. K. Gusti, "Implementation of convolutional neural network algorithm (ResNet-50) for benign and malignant skin cancer classification," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 3, pp. 984–992, 2024, doi: 10.57152/malcom.v4i3.1398.
[26] E. Setia Budi, A. Nofriyaldi Chan, P. Priscillia Alda, and M. Arif Fauzi Idris, "Optimization of machine learning model for image classification and prediction using convolutional neural network algorithm," *Media Online*, vol. 4, no. 5, p. 509, 2024, [Online]. Available: https://djournals.com/resolusi
[27] F. Rauf *et al.*, "Artificial intelligence assisted common maternal fetal planes prediction from ultrasound images based on information fusion of customized convolutional neural networks," *Front. Med.*, vol. 11, no. October, 2024, doi: 10.3389/fmed.2024.1486995.
[28] X. Yang, S. Yu, and W. Xu, "Enhanced convolutional neural networks for improved image classification," 2025, [Online]. Available: http://arxiv.org/abs/2502.00663
[29] B. Faye, M. Lebbah, and H. Azzag, "Supervised batch normalization," 2024, [Online]. Available: http://arxiv.org/abs/2405.17027
[30] L. Zhao and Z. Zhang, "A improved pooling method for convolutional neural networks," *Sci. Rep.*, vol. 14, no. 1, pp. 1–23, 2024, doi: 10.1038/s41598-024-51258-6.
[31] X. Kong, X. Liu, J. Gu, Y. Qiao, and C. Dong, "Reflash dropout in image super-resolution," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2022-June, pp. 5992–6002, 2022, doi: 10.1109/CVPR52688.2022.00591.
[32] M. M. Taye, "Theoretical understanding of convolutional neural network: concepts, architectures, applications, future directions," *Computation*, vol. 11, no. 3, 2023, doi: 10.3390/computation11030052.
[33] M. Lin, Q. Chen, and S. Yan, "Network in network," *2nd Int. Conf. Learn. Represent. ICLR 2014 - Conf. Track Proc.*, pp. 1–10, 2014.
[34] Sutedi, M. Royan, F. Maulana, M. Agarina, and A. Suryadi, "Brain tumor detection on magnetic resonance imaging using deep neural network," vol. 7, no. 1, 2023.
[35] L. A. Yates, Z. Aandahl, S. A. Richards, and B. W. Brook, "Cross validation for model selection: A review with examples from ecology," *Ecol. Monogr.*, vol. 93, no. 1, pp. 1–24, 2023, doi: 10.1002/ecm.1557.
[36] V. M. Patro and M. R. Patra, "A novel approach to compute confusion matrix for classification of n-class attributes with feature selection," *Trans. Mach. Learn. Artif. Intell.*, vol. 3, no. 2, 2015.
[37] L. Raditya, "Metrik evaluasi untuk klasifikasi," *Medium*, 2022. https://luthfiradity.medium.com/evaluating-classification-model-a2e19a50acee (accessed Feb. 02, 2025).
[38] K. S. Nugroho, "Confusion matrix untuk evaluasi model pada supervised learning," *Medium*, 2019.
[39] M. E. Klontzas, G. Kalarakis, E. Koltsakis, T. Papathomas, A. H. Karantanas, and A. Tzortzakakis, "Convolutional neural networks for the differentiation between benign and malignant renal tumors with a multicenter international computed tomography dataset," *Insights Imaging*, vol. 15, no. 1, 2024, doi: 10.1186/s13244-023-01601-8.
[40] K. Faryna, J. van der Laak, and G. Litjens, "Automatic data augmentation to improve generalization of deep learning in H&E stained histopathology," *Comput. Biol. Med.*, vol. 170, no. October 2023, p. 108018, 2024, doi: 10.1016/j.compbiomed.2024.108018.
[41] E. Goceri, *Medical image data augmentation: techniques, comparisons and interpretations*, vol. 56, no. 11. Springer Netherlands, 2023. doi: 10.1007/s10462-023-10453-z.
[42] O. Rainio and R. Klén, "Comparison of simple augmentation transformations for a convolutional neural network classifying medical images," *Signal, Image Video Process.*, vol. 18, no. 4, pp. 3353–3360, 2024, doi: 10.1007/s11760-024-02998-5.
[43] L. Huang, J. Qin, Y. Zhou, F. Zhu, L. Liu, and L. Shao, "Normalization techniques in training DNNs: methodology, analysis and application," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 8, pp. 10173–10196, 2023, doi: 10.1109/TPAMI.2023.3250241.
[44] C. Ozdemir, Y. Dogan, and Y. Kaya, "A new local pooling approach for convolutional neural network: local binary pattern," *Multimed. Tools Appl.*, vol. 83, no. 12, pp. 34137–34151, 2024, doi: 10.1007/s11042-023-17540-x.
[45] Z. Liu, Z. Xu, J. Jin, Z. Shen, and T. Darrell, "Dropout reduces underfitting," *Proc. Mach. Learn. Res.*, vol. 202, pp. 21715–21729, 2023.
[46] D. T. Tran, N. Shimada, and J. H. Lee, "Triple-sigmoid activation function for deep open-set recognition," *IEEE Access*, vol. 10, no. June, pp. 77668–77678, 2022, doi: 10.1109/ACCESS.2022.3192621.
[47] GeeksforGeeks, "Why an Increasing Validation Loss and Validation Accuracy Signifies Overfitting?," *GeeksforGeeks*, 2023. https://www.geeksforgeeks.org/why-an-increasing-validation-loss-and-validation-accuracy-signifies-overfitting (accessed Feb. 02, 2025).
[48] Lutz Prechelt, "Early stopping - but when?," no. March 2000, 2015, doi: 10.1007/3-540-49430-8.
[49] M. Valdenegro-Toro and M. Sabatelli, "Machine learning students overfit to overfitting," *Proc. Mach. Learn. Res.*, vol. 207, pp. 46–51, 2022.
[50] X. Ying, "An overview of overfitting and its solutions," *J. Phys. Conf. Ser.*, vol. 1168, no. 2, 2019, doi: 10.1088/1742-6596/1168/2/022022.