EfficientNet-Based Melanoma Classification with CBAM Attention and Monte Carlo Dropout for Robust Uncertainty Estimation

Soujenya Voggu, Shadab Siddiqui*, Shahin Fatima
Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,
Hyderabad-500075, Telangana, India

Abstract—**Recent developments in deep learning have demonstrated tremendous potential for enhancing medical picture classification tasks, particularly for the detection of skin malignancies like melanoma. However, it is still a huge challenge to guarantee high accuracy, reliability, and interpretability in real clinical settings. This study attempted to resolve these issues by proposing a novel approach to melanoma detection, by employing diverse techniques such as the Convolutional Block Attention Module (CBAM), binary focal loss, and Monte Carlo Dropout (MC Dropout) for uncertainty estimation. The CBAM attention module was inserted to help the network focus on important features of images, and focal loss was applied to solve class imbalance and encourage learning from hard samples. MC Dropout was used to achieve an uncertainty estimate in the test set, and thus, more reliable and interpretable predictions. The approach was implemented with a pre-trained deep CNN called EfficientNetB4 as the backbone and trained on a large melanoma dataset, which is separated into training sets, test sets, and validation sets in order to test the performance. Model evaluation was performed using accuracy, precision, recall, F1-score, and AUC, resulting in 0.95 for accuracy, whereas the AUC value is 0.98. Furthermore, the uncertainty estimate made a clearer decision-making, and the interpretability was crucial when used as a clinical task model. These results highlight the necessity to combine attention mechanisms, task-specific loss terms, and uncertainty quantification for building accurate and interpretable AI in medical domains. The study prototype has the potential for improving the detection of early-stage melanoma and provides useful guidance to future AI-based healthcare services.**

Keywords—Deep learning; CNN; accuracy; CBAM; EfficientNetB4

I. Introduction

Melanoma is an aggressive skin cancer and has continued to be a leading cause of death from cancer worldwide [1]. Melanoma – which originates in the pigment-producing cells of the skin called melanocytes – can spread rapidly to other organs unless it is caught and treated early. Early detection of melanoma contributes significantly to the survival rates of patients and is, therefore, a critical area of research in medical imaging and diagnostics [2]. Diagnosis of melanoma is usually through clinical examination and histopathology, which are associated with human error and variability. Due to the visual aspect of the disease, automatic systems and methods, which make use of image analysis, are a central part of new scientific progress on accurate melanoma detection.

Recent advances in machine learning, particularly deep learning, have demonstrated significant potential for automated melanoma classification based on skin lesion images [3]. Convolution neural network (CNN) has been known as a successful technique for image classification, where it can learn and abstract features automatically from visual data. Melanoma classification has been tackled using different methods, with the use of diverse CNN architectures to enhance accuracy. However, there are still many issues, such as the limited labeled datasets in diverse domains, the differences between the quality levels of images and patterns of skin lesions complexity [4]. Hence, emerging models capable in their ways to go beyond these limitations and make more appealing predictions (in terms of reliability, interpretability and robustness) are always needed.

The problem of handling the uncertainty relating to the medical image analysis is one of the primary concerns for melanoma classification [5]. Despite their great power, deep learning models are typically unable to make predictions while also estimating the uncertainty of these predictions. Failure to estimate such uncertainty can lead to bad decisions in a clinical setting when missclassification is expensive [6]. Also, deep learning based algorithms are prone to overfitting when there is limited The study proposed a new deep learning framework for melanoma recognition, where the CBAM attention mechanism, focal loss and Monte Carlo Dropout (MC Dropout) were integrated to obtain uncertainty estimation. The incorporation produced a remarkable improvement in the performance and reliability. With the help of these sophisticated methods, our model could focus on important attributes and handle the issue of class imbalance to generate reliable predictions with uncertainty. With an accuracy of 95 and an AUC of 0.98, the model also has great potential in future real clinical applications. The novelty of our method lies in the usage of attention mechanism, focal loss and uncertainty estimation, optimizing them all simultaneously. This means to bring interpretability and robustness at the same time so that our model can generalize well for difficult tasks such as melanoma detection, training data or unbalanced data. All of these are reminiscent of the need to come up with methods that improve model accuracy and also provide uncertainty estimates' which is a necessary condition for well-calibrate decision support

*Corresponding author.

systems able to offer robustness and generalizability in clinical usage.

Another is to determine what are the most relevant parts in an image, particularly when diagnosing melanomas [7]. Even if the CNNs are good feature extractors, this learning does not exploit spatial relationships and contextual information among the entities in an image. It can result in a mistake with grouping, particularly if the melanoma lesion is not readily visible or partly covered. The Convolutional Block Attention Module (CBAM): This has proved beneficial to enhance the performance of CNNs, so they attend more on a most important part of an image. By utilizing an attention mechanism, CBAM makes networks pay more attention on informed regions to increase the classification accuracy and interpretability.

To tackle the above difficulties, a cost-effective and dynamic solution powered by an EfficientNet with CBAM attention-based model, incorporated with Monte Carlo Dropout (MCD) for uncertainty estimation. Derivation of network: EfficientNet has been chosen as it has strong and robust accuracy and computation efficiency according to retrospective training, testing using images regarding melanoma [8]. MCD can naturally give an approximate measure of prediction uncertainty, while the integration of CBAM may assist the network in attending to saliency parts on input images. This synergistic relationship leads to a stronger identification of melanoma, which in turn is good-performing (in terms of accuracy) and provides uncertainty, thereby allowing its application in the clinic where decisions are crucial.

II. Literature Survey

S. Nazari et al. [9] stressed the relevance of diagnostic models for real-world usage, which include smartphone based dermoscopyin order to enhance the diagnosis in rural regions. As per the ISIC2020 challenge, their approach demonstrated comparable outcomes to the first-place model while outperforming second and third places. These models established an appropriate compromise between accuracy and computational efficiency by using 98 % fewer parameters, thereby making them well suited for real-time deployment in resource constrained environment and increasing the reach of accessible healthcare.

A hybrid framework was proposed by P. K. Veni et al. [10] that integrates feature extraction from VGG16 with CBAM and classification with Caps Net. The proposed model was tested using augmented and non-augmented datasets, thereby displaying high performance results up to 100 % precision, 99 % accuracy, and an F1-score.

Skin-GAB, a deep learning based scheme, is proposed by J. Chen et al. [11] for classifying pigmented skin diseases. This approach makes use of augmentation, segmentation, network fusion, and the GAB mechanism for enhancing classification by highlighting essential characteristics. Skin GAB has enhanced the accuracy 2.89 % as compared to the previous model, projected to continue contributing to future diagnosis and managing pigmented skin illness.

A most dangerous type of skin cancer, focusing melanoma, was proposed by G. Dogan et al. [12]. If diagnosed early, it can be treated effectively.

Convolutional neural networks (CNNs), a popular deep learning tool, have shown great promise in accurately detecting melanoma. By integrating attention mechanisms with CNNs, their accuracy can be further enhanced. However, previous models for melanoma detection have not utilized attention mechanisms effectively. In order to help researchers use attention mechanisms effectively, this study investigates how they affect CNN performance. The authors investigated seven distinct attention mechanisms using a base CNN model and compared the outcomes.

The C3BAM-XAI model of CNN was proposed by M. J. Abbas et al. [13], integrating CBAM and explainable AI. It addresses data imbalance through augmentation and uses CBAM’s attention modules for better feature extraction. Hyperparameters are optimized with Nadam for smoother training. The model, tested on the PD Kaggle dataset, achieved 93.33% accuracy by B. Mittal [14]. Ablation trials demonstrate that CBAM will enhance the interpretability and accuracy of the model, making it dependable for clinical usage, particularly for Parkinson’s disease detection. This study highlights the expanding significance of deep learning approaches in improving medical diagnosis.

CACBL-Net, a lightweight deep CNN, was proposed by R. Agrawal et al. [15], tailoring portable diagnostic devices like smartphones, thereby addressing limitations of data imbalance and computational capacity. The study made use of focal loss for handling imbalance in the dataset and the Monte Carlo dropout method for uncertainty estimation to improve the robustness of diagnostic networks.

The network incorporates channel attention to enhance feature extraction and employs an adaptive class-balanced focal loss to prioritize complex cases while reducing the influence of simpler ones, B. Mittal [16]. On the HAM-10000, PAD-UFES-20 and MED-NODE datasets, CACBL-Net reached sensitivities of 90.60%, 91.88% and 91.31% with prediction times ranging from 0.006 to 0.011 s per case, respectively. These performances are superior to other models that prove the potential and CACBL-Net for skin lesion diagnosis using a mobile device with low computational cost.

Z. Ji et al. [17] introduced EFAM-Net as a new architecture for the classification of skin lesions. The low-level information of color and texture is learned by inserting an Attention Residual Learning ConvNeXt (ARLC) block in the shallow part, and as the depth increases, it is removed to employ a deeper layer with the Parallel ConvNeXt (PCNXt) block by M.K. Amber [18]. To enhance the feature fusion in multiple scales, a Multi-scale Efficient Attention Feature Fusion (MEAFF) block is proposed. We tested our method on ISIC 2019, HAM10000 and a private dataset and the experiment results showed it performed satisfactorily with classification accuracies of 92.30%, 93.95%, and 94.31%.

Y. Jia et al. [19] proposed a medical image classification approach, which adopted a contour processing attention mechanism to increase its accuracy through focusing on key regions. The process comprises a step of photo converting and linearizing images, as well as contour generation.

The original gray-scale picture is convolved with these contours, creating a row-column feature map that can be further sharpened by means of point-wise multiplication by A. Siddiqui [20]. The resulting image is input to a residual network for classification. Experiments on three medical image datasets indicate noticeable enhancements are achieved with respect to accuracy, F1 score, and Kappa score. This technique also demonstrates value in other areas such as remote sensing and vehicle image recognition.

III. Proposed Model

A. Input Layer

The model receives data through the input layer. In this instance, the input layer is set up to accept photos of shape (224, 224, and 3), which means the images have three color channels (RGB) and a dimension of 224 by 224 pixels. In order to ensure the model correctly processes the data, this layer specifies the form of the input data that will be fed into the neural network.

$$Input = \mathbb{R}^{H \times W \times C} \tag{1}$$

where,

H stands for height, W for width, and C for the number of channels (3 for RGB) in the picture. This layer does not do any calculation; it only takes in the input.

B. EfficientNetB4

The foundation of the model for feature extraction is a pre-trained deep CNN called EfficientNetB. It belongs to an efficient net family, which is renowned for its excellent computational cost and parameter size efficiency. The top classification layers are eliminated by setting include top=False, allowing the model to focus on extracting relevant features from the image data. The model learns superior initial feature representations thanks to the pre-trained weights from ImageNet, which lessens the need for intensive training from the beginning. Similar to other convolutional neural networks, theEfficientNetB4 model processes the input image in several ways:

Convolution Operation:

$$Output = Conv(X, W, b) = (X * W) + b \tag{2}$$

where,

X is the input tensor.

Wis the filter (kernel).

b is the bias.

∗represents the convolution operation.

Activation: Following the convolution, an activation function such as ReLU is applied:

$$Output = ReLU\big(Conv(X, W, b)\big) \tag{3}$$

EfficientNetB4 uses advanced blocks like MBConv, Squeeze-and-Excitation (SE) blocks, and depth-wise separable convolutions, but the basic operation is a convolution followed by activation.

C. Dense Layers (Channel Attention)

The feature maps are transformed by the channel attention module dense layers following pooling operations. By using the ReLU activation function, the first dense layer reduces the number of channels by a factor of the ratio (usually set to 8). The feature map is compressed throughout this process, thereby allowing the network to learn more succinct representations. By returning the channel size to its initial value, the second dense layer produces an improved feature map that may be integrated with the attention mechanism.

First Dense Layer:

$$Output_1 = ReLU\left(\frac{X}{Ratio}\right) \tag{4}$$

where,

- X is the input feature map from the previous layer.
- Ratio is a hyperparameter (often set to 8).

Second Dense Layer:

$$\mathrm{Output}_2 = \mathrm{W.Output}_1 + \mathrm{b}_1 \tag{5}$$

where,

- Wis the weight matrix.
- b is the bias vector.
- The dense layer performs a matrix multiplication followed by the addition of bias

D. GlobalAveragePooling2D

GlobalAveragePooling2D calculates the average of every feature map throughout the full spatial dimension (including height and width). It makes the output size uniform across various input images by reducing the spatial dimension to a single value per feature map. By ensuring that each feature map is summed up by its average value, this pooling procedure enables the model to incorporate global context and exclude spatial details that are less crucial for the job. The average of each feature map of all spatial locations is calculated via global average pooling:

$$Output_{avg}[c] = \frac{1}{H \times W} \sum_{h=1}^{H} \sum_{w=1}^{W} X[h, w, c] \tag{6}$$

where,

- Hand W are the height and width of the feature map.
- c is the index for the channels.
- The output is the average value of all the pixels in each channel of the feature map.

E. Reshape Layer

The reshape layer is used to transform the output of the global average pooling operation into a shape that corresponds to the layers it follows. In this instance, it ensures that the tensor has a single spatial dimension while retaining the channel information by reshaping the pooled tensor to a shape of (1,1, channel). As a result, the feature map can be processed by the thick layers that follow, which are built to deal with a specific dimensionality.

$$\text{Reshaped Output} = \text{Reshape}\big(\text{Input}, (1,1, \text{Channels})\big) \quad (7)$$

The reshape operation doesn't modify the underlying data but changes the dimensions of the tensor to ensure compatibility with subsequent layers. The result is a tensor of shape (1,1,Channels), where each channel contains one averaged value.

F. GlobalMaxPooling2D

Similar to Global Average Pooling, Global Max Pooling calculates the maximum value for each feature map rather than averaging the values. The notable feature of each feature map is highlighted using this technique. For tasks that depend on identifying strong patterns, such as object recognition, it captures the most prominent spatial characteristic.

Global Max Pooling computes the maximum value for each feature map:

$$\text{Output}_{\text{max}}[c] = \max_{h,w} X[h, w, c] \quad (8)$$

where,

- [h,w,c] is the value at spatial location h,w, and channel c.
- This outputs the maximum value for each feature map, providing a more extreme form of feature aggregation compared to averaging.

G. Dense (After MaxPooling)

After max pooling, the features are processed by the same dense layers that were used for average pooling. These layers help refine the pooled features by applying the same transformations (reduction in channels and restoration) as seen with average pooling. This dual processing allows the model to consider both the maximum and average features extracted from the input image.

$$\text{Output} = W \cdot X + b \quad (9)$$

where,

- W is the weight matrix,
- X is the input feature map from the pooling operation.
- b is the bias term.

This is a standard dense layer that performs matrix multiplication and bias addition.

H. Add Layer

The results from both average and max pooling are then aggregated (Add layer). This kind of addition has more rich feature representation by using complementary information involved in the two pooling types. With a Concatentity set, the model perhaps can more easily focus on both the prototype categories and the salient cues in the image.

$$\text{Output} = \text{avg\_pool} + \text{max\_pool} \quad (10)$$

This layer performs the addition of the output from average pooling and max pooling. Each result element is the sum of its two feature map elements.

I. Activation (Sigmoid)

The Activation layer, applying the sigmoid activation function, is used in order to squash attributes of two branches into [0, 1] value softening. This yields a tensor which can be employed as the attention weights. The attention map generated by this activation enables the network to focus on or ignore some feature channels according to their relevance to the task.

$$\text{Output} = \frac{1}{1+e^{-X}} \quad (11)$$

The sigmoid function normalizes values between 0 and 1, making it ideal for binary classification tasks, where the output is interpreted as a probability.

J. Multiply (Channel Attention)

The attention mechanism for the input features is added via the multiply layer. The input feature map is reweighted with the attention map of previous layers through element-wise multiplication. This operation enhances some objects while reducing others over the input. This attention mechanism successfully allows our network to pay more attention to the most essential parts of the input image.

$$\text{Output} = X \times A \quad (12)$$

The attention map A, produced by applying sigmoid activation to the attention mechanism, is element-wise multiplied with input feature map X. The operation highlights the most salient features while suppressing irrelevant ones.

K. Lambda Layer (Spatial Attention)

The Lambda layer average-pools the input feature map over its spatial dimensions. By means of spatial dimension reduction, it yields a tensor that characterizes the distribution of global features in the image. This enables the model to obtain context throughout the entire image, where contextual information may be important for feature relationships (e.g., object detection and segmentation).

$$\text{Output}_{\text{avg}} = \frac{1}{H\times W}\sum_{h=1}^{H}\sum_{w=1}^{W} X[h, w, :] \quad (13)$$

$$\text{Output}_{\text{max}} = \max_{h,w} X[h, w, :] \quad (14)$$

In the Lambda layer, we calculate the mean and max pixel values for each feature map over spatial dimensions. This permits the spatial information to be 'out-of-context', thus facilitating attention-guiding.

L. Concatenate Layer

The results of average and max pooling are concatenated by the concatenate layer in the channels dimension. This concatenation results in a more enriched feature map, including the average and most discriminative spatial features, which could be utilized by the model for both average-based and peak-based aspects at some layers ahead. It facilitates the network to construct a better representation for images by modeling different types of feature information.

$$Output = Concatenate(avg\_pool, max\_pool) \quad (15)$$

The average and max pooled features are concatenated at channel level. This operation doubles the number of channels as both types of pooling information is merged into one tensor.

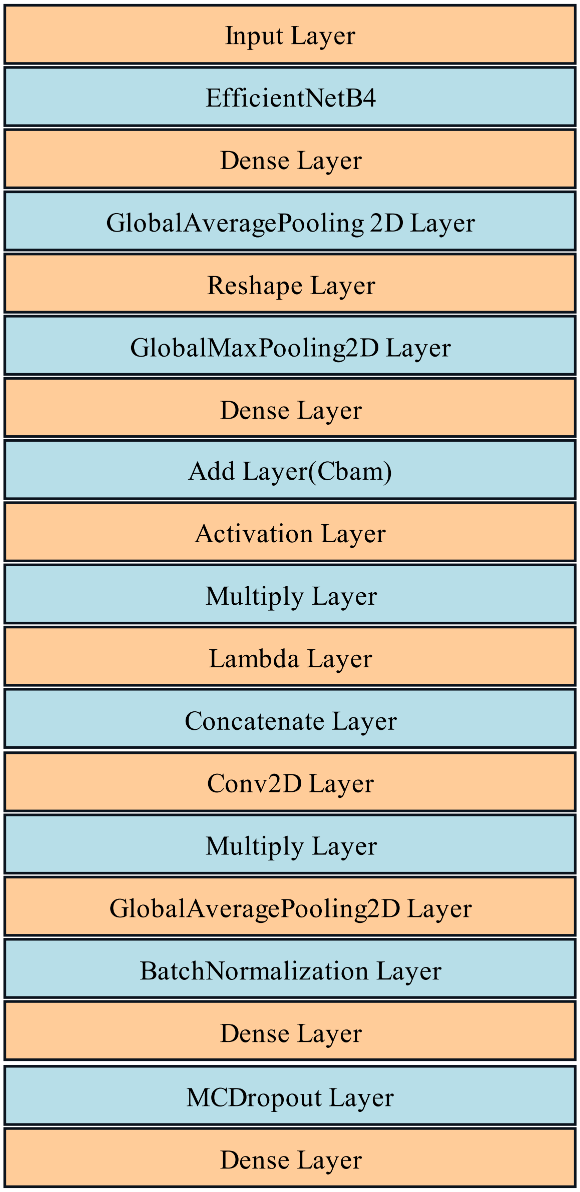

Fig. 1. Proposed model architecture.

Fig. 1 demonstrates the architecture of the proposed hybrid CNN model by combining EfficientNetB4 with CBAM for enhanced feature extraction. The layered structure in the figure illustrates components like convolution, pooling and attention mechanisms.

M. Conv2D (Spatial Attention)

A convolutional filter is then used by the Conv2D layer on the concatenated feature map to generate a spatial attention map. The layer is trained to pay attention to parts of the image, through learning of spatial features, so as to highlight the most relevant parts. Kernels size and strides are determined to maintain the spatial relationship in the image as well as computational efficiency. The sigmoid activation makes the attention values range within 0 and 1, controlling how much each spatial location needs to be weighted.

$$Output = Sigmoid\big(Conv2D(Concat, Filters, Kernel\ Size)\big) \quad (16)$$

A convolution operation is performed over the concatenated feature maps and fed into a sigmoid function for obtaining the spatial attention map, which emphasize the region at a spatial level that model should pay attentions to.

N. Multiply (Spatial Attention)

Similar to the channel attention mechanism, the multiply layer applies the spatial attention map to the input features. The input feature map is multiplied by the spatial attention map, allowing the model to focus on the important spatial regions while ignoring less relevant ones. This helps the network learn which regions in the image are critical for the task at hand.

$$\text{Output} = X \times A_{\text{spatial}} \quad (17)$$

The input feature map X is element-wise multiplied by the spatial attention map A_$_{spatial}$, highlighting the most important spatial regions. Which focuses on the most relevant spatial regions.

O. Global Average Pooling2D (Post Attention)

To summarize all feature maps, GlobalAverage Pooling 2D is applied after the application of CBAM attention mechanism. In order to prepare the output for thick layers, this pooling step reduces the spatial dimensions of each feature map to a single value. After using the attention mechanism, this step guarantees that network is taking into account global features in feature map.

$$\text{Output}_{\text{avg}} = \frac{1}{H\times W}\sum_{h=1}^{H}\sum_{w=1}^{W} X[h,w,:] \quad (18)$$

Global average pooling is applied again after the attention mechanism to summarize the entire feature map into a single value per feature map.

P. Batch Normalization

Batch Normalization helps to stabilize and speed up training by normalizing activation of neurons throughout the mini-batch. It guarantees the consistency of activation distribution by minimizing internal covariate shift. Furthermore, batch normalization improves the generalization of model and training process efficiently by acting as a regularization strategy.

$$\text{Output} = \gamma\left(\frac{X-\mu}{\sqrt{\sigma^2+\epsilon}}\right)+\beta \quad (19)$$

Batch normalization normalizes the activations by subtracting the batch mean μ and dividing by the batch standard deviation σ. During training, the parameters γ(scale) and β(shift) are learned to adjust the normalized values.

Q. Dense (Fully Connected Layer)

The Dense layer of size 256 propagates the feature map through a fully connected transformation. It learns the high abstractions of stance properties (e.g., training structures) and is called “ReLU” which introduces non-linearity into the model. This layer is important so that the model can learn complex patterns in between before outputting something simple. L2 regularization is introduced to avoid overfitting by punishing excessive weights.

$$\mathrm{Output} = \mathrm{ReLU}(W.X + b) \tag{20}$$

This dense layer performs a matrix multiplication between the input and weight matrix W, adds the bias term b, and applies the ReLU activation function.

R. MCDropout

The Dropout layer is a dropout layer with a custom forward pass that remains active during test. This is in contrast to traditional dropout, which is employed only at training time and thus cannot measure the uncertainty during test. This is especially valuable for uncertainty estimation tasks, where the interpretability of the model's confidence in prediction is key. Through the activation of dropout during inference, the model is able to produce a wider diversity of predictions, leading to a more reliable measure for uncertainty.

$$\mathrm{Output} = \mathrm{Dropout}(X, p = 0.5) \tag{21}$$

The MCDropout layer ensures that dropout is active during both training and inference, which allows the model to generate multiple stochastic forward passes and estimate uncertainty in predictions.

S. Dense (Output Layer)

The final Dense layer is the output with a single unit for binary classification and sigmoid activation to make it suitable for your prediction task. Finally, the output of the sigmoid gives a value between 0 and 1, which one might interpret as being the probability that your input is in the positive class. This layer is the final decision point of the network, and observations fit to the model are transformed into a modeled prediction.

$$\mathrm{Output} = \mathrm{Sigmoid}(W \cdot X + b) \tag{22}$$

The final output layer calculates the probability of the positive class (binary problem) with a sigmoid activation. The output is a score in the range of [0, 1], which indicates the probability that the input sample falls into the positive class.

The novel changes in the model could improve both performance and interpretability on a melanoma deep learning classifier. The contribution lies in integrating the CBAM (Convolutional Block Attention Module) that works with channel-wise and spatial attention. As a result, this allows the model to pay attention on most informative regions of the input image to improve feature extraction, which will automatically emphasize useful parts. Additionally, the model has a binary focused loss, which addresses class imbalance by emphasizing hard negatives more. The Monte Carlo dropout introduces uncertainty estimation during interference, which can be used to measure the degree of prediction confidence and reveal information about the dependability of the model. These enhancements are intended to maintain a reliable and understandable method of melanoma classification, which also improves the generalization of model performance and reduces overfitting.

Advanced data augmentation and specially designed callbacks like ReduceLROnPlateau and EarlyStopping, which aid in learning optimization and prevent overfitting, are some further contributions of the suggested model. The model uses EfficientNetB4 as a backbone to extract features, making it possible for the model to achieve high accuracy with computational cost under control. We train the model in multiple stages with heavy data augmentation like rotation, zoom and flip to be able to generalize well to unseen data. In the evaluations, besides these standard measures (accuracy, precision, recall and F1-score), we measure uncertainty using Monte Carlo sampling. This perspective offers a broader interpretation of model behavior that can guide confidence in clinical decisions.

IV. Experimental Results

The results produced by the method developed in this work during the current simulation studies are reviewed in this section of the study.

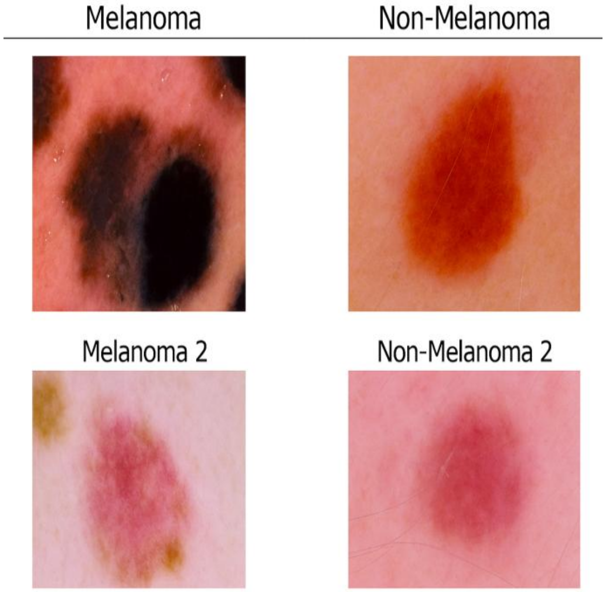

Fig. 2. Sample images for melanoma and not melanoma classification.

Fig. 2 shows the sample images taken for classifying Melanoma and Not Melanoma. The melanoma dataset was used for these experiments [17]. The data preprocessing techniques described earlier were also applied to this dataset in our work. The proposed work made use of ISIC 2019 [21], [22] and HAM10000 [23] public datasets available freely online on the International Skin Imaging Collaboration Challenge.

The plot in Fig. 3 shows the training and validation accuracy across 50 epochs. The training accuracy curve, indicated by the blue line, gets very jagged with vivid spikes and ascents. This might reflect a tendency to overfit, implying that the model is making its predictions too closely follow the training set (as opposed to other data). On the other hand, the validation accuracy (green) moves more steadily on small steps of improvement, which means better generalization to unseen data. The saturation of the generalization g between the two accuracy curves suggests that the depth-3DIHNN fails to generalize well, and this problem would be alleviated by techniques such as model fine-tuning or regularization.

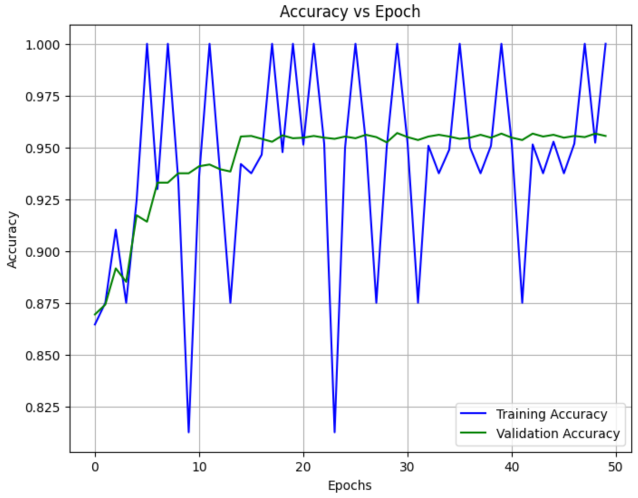

Fig. 3. Training and validation accuracy over epochs.

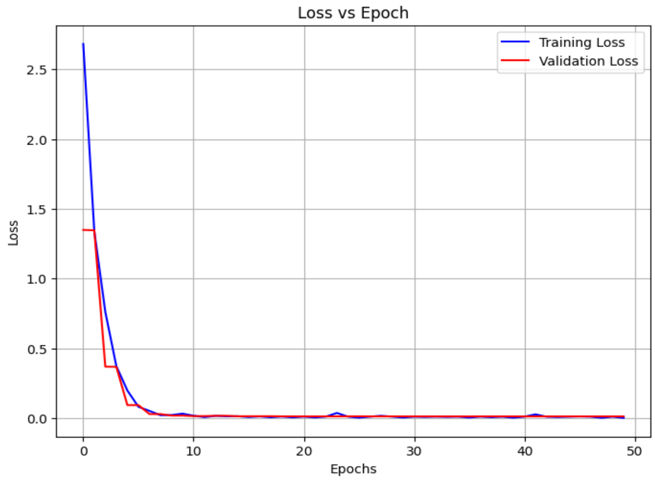

Fig. 4. Training and validation loss over epochs.

The above plot in Fig. 4 visualizes how loss changes over time with epochs as the independent variable. Training loss is the purple line, validation loss is the red line. Both the losses start large but decrease rapidly, indicating that the model makes good progress early in training. After this drop, there is some breakage, and the losses rapidly go lower in later epochs. Training and Validation loss have nearly no split, it means the model is not overfitting or underfitting. As the model is able to achieve low loss values, it suggests that the model can learn well.

The Receiver Operating Characteristic (ROC) curve, shown in Fig. 5, is a popular measure of the performance of a binary classifier is depicted in the above figure. The relationship of the true positive rate (TPR) versus the false positive rate (FPR) is also plotted with different threshold values. The blue line is shooting up towards high TPR values as FPR rises – this means that the model can differentiate between positive and negative classes well. The curve is shifted towards the upper left corner (optimal performance) with high sensitivity and specificity. The dashed diagonal line is a random classifier which have same TPR as it has FPR for any threshold. The ROC curve is also well above this diagonal, so it shows that the model beats random guessing. Area Under the Curve (AUC) is probably close to 1, indicating strong classification.

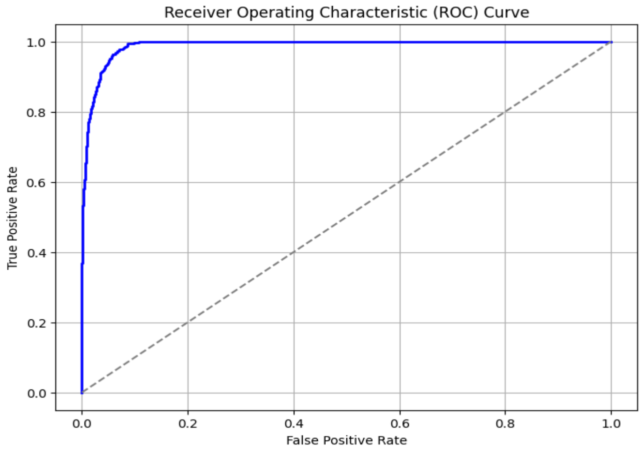

Fig. 5. Receiver Operating Characteristic (ROC) curve.

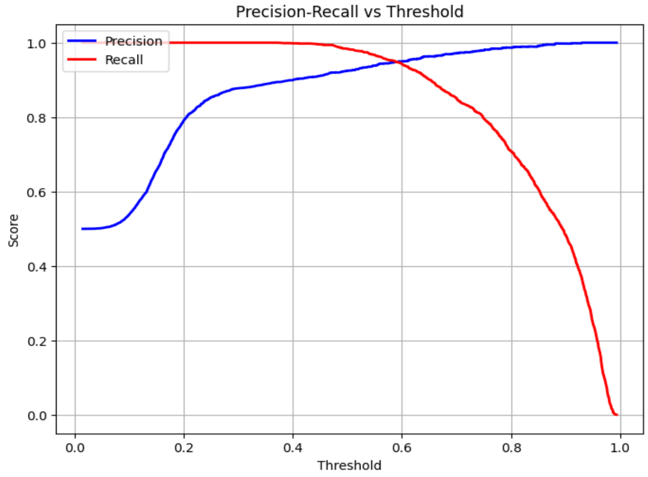

Fig. 6. Precision and recall vs. Threshold.

The above Fig. 6 shows the Precision-Recall curve against the classification threshold. The blue curve corresponds to precision, while the red one indicates recall. As we set the '+' threshold higher and higher, our precision improves drastically, approaching 1, which means that the model becomes increasingly confident in its positive predictions to yield high overall accuracy. But recall (the red curve) is decreasing as precision increases. This is a simple case of trade-off when we tune the threshold - recall increases (hits more true positives) and precision decreases (hits false positives). The feature space gives a geometrical picture of this trade-off, and allows to choose the best threshold with respect to the precision-recall trade-off according to the particular characteristics of the problem (e.g., more emphasis on precision or recall).

The Cumulative Gain Curve shown in Fig. 7 is something that will help you when it comes to evaluating the performance of a classifier for ranking problems. The curve displays the cumulative gain against the percentile of the samples. With the number of samples, the cumulative gain increases as well and tends to 1. This indicates that if 80% of samples are retained by taking those after the 20th row, a larger percentage of the positive class is covered. The steep rise at the start of the curve shows that the model can quickly identify many relevant samples, which could be advantageous for systems such as targeted marketing or fraud detection. A perfect curve would indicate a sharp steep at the beginning, followed by a plateau, indicating that most positive samples are correctly included in top ranked portion of the test.

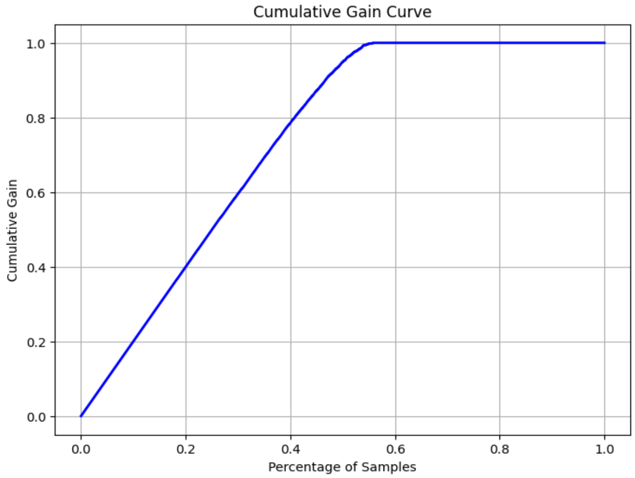

Fig. 7. Cumulative gain curve.

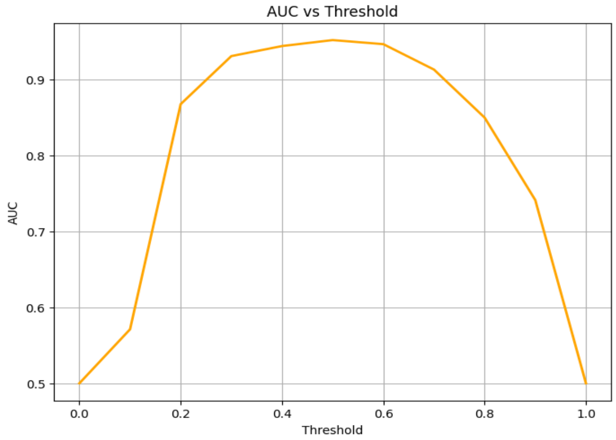

Fig. 8. AUC vs. Threshold.

Fig. 8 shows how the Area Under the Curve (AUC) and classification level relate. Plot of AUC as a function of the threshold defining positive and negative samples. At first, as the threshold increases substantially from 0 to about 0.3, AUC rapidly grows and it is close to the maximum value of almost 1.0, showing that the model has great classification performance at this threshold grain highly balanced class separation. More importantly, the AUC starts to decline when this threshold becomes larger, implying that with higher threshold values, the model down-weights positive samples and accordingly its capability of distinguishing positives from negatives weakens. Such behavior suggests that an optimal threshold can be found where AUC is the largest for indicating the threshold that results in the best classification performance.

This decline following the peak is expected due to the well-known trade-off between precision and recall with threshold.

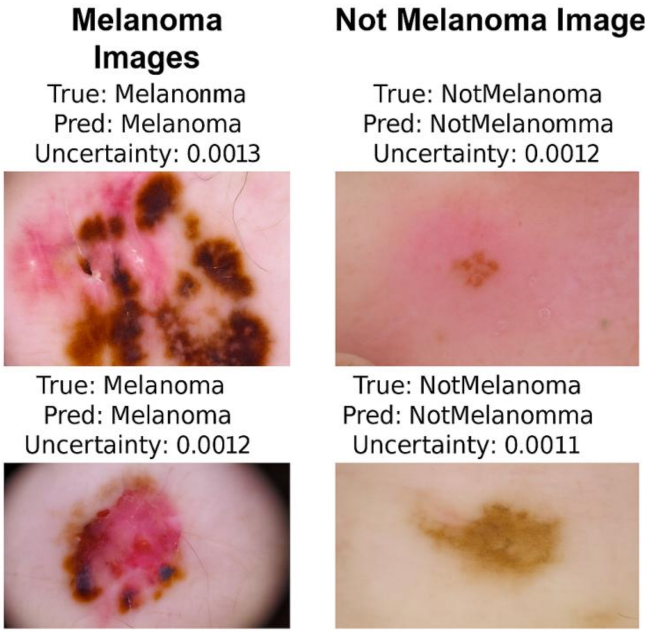

Fig. 9. Classification of melanoma and not melanoma images with prediction confidence.

The picture in Fig. 9 represents the classification between Melanoma and Not Melanoma images and model predictions with respective uncertainty values. The left column shows Melanoma cases that are correctly predicted as Melanoma, with uncertainty of 0.0013 and 0.0012, so the model is more certain in these predictions. The right column shows Not Melanoma images classified as such with even lower uncertainty values (U = 0.0011 and U = 0.0012). These very low levels of uncertainty indicate that the model is doing an excellent job accurately placing examples into one of two categories, and therefore can be thought to have high certainty in its predictions.

TABLE I. CLASSIFICATION REPORT

	Precision	Recall	F1-Score
Melanoma	0.98	0.91	0.95
Not Melanoma	0.92	0.98	0.95
Total Accuracy	0.95		

The above Table I is the classification report for a model that classifies between two labels, Melanoma and Not Melanoma. In the case of Melanoma, the model has high precision (0.98), which means that 98% are the predicted positive cases are comparable. Its recall (0.91) is a bit lower, which means that 91% of the actual positive cases have been recognized. Melanoma achieves F1-Score 0.95, balancing precision and recall. For Not Melanoma, it is 0.92 precision and a high recall of 0.98 indicates that the model accurately detects true positives at a high rate. The Not Melanoma F1-Score: 0.95 as well. The suggested model, making use of Efficient Net using CBAM Attention and Monte Carlo Dropout for uncertainty estimation and the model's general accuracy of 0.95 indicates a strong performance for the two classes. The balanced F1-Scores and high accuracy demonstrate that the model performs well in detecting both conditions without significant bias.

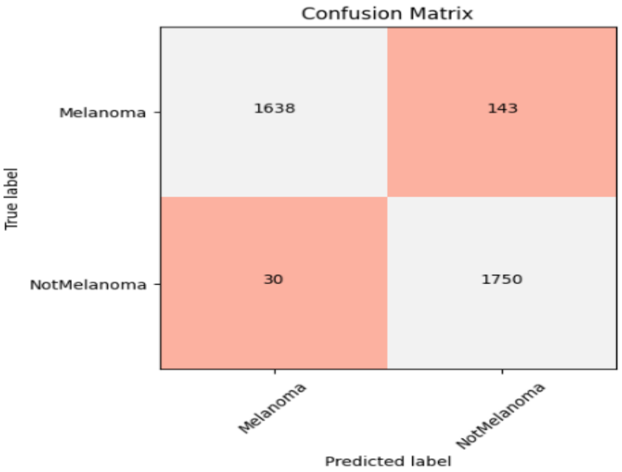

Fig. 10. Confusion matrix for melanoma classification.

Fig. 10 above shows the confusion matrix for a binary classification model distinguishing between Melanoma and Not Melanoma. Out of 2816 instances, the model correctly identified 1638 cases of Melanoma (True Positives) and 1750 cases of Not Melanoma (True Negatives). However, it made errors by incorrectly predicting 143 cases of Not Melanoma as Melanoma (False Positives) and 30 cases of Melanoma as Not Melanoma (False Negatives). While the matrix reflects the model's strong performance in accurately classifying Not Melanoma (with high True Negatives), it also reveals areas for improvement, particularly in reducing the False Positive and False Negative predictions.

TABLE II. COMPARATIVE ANALYSIS

Model Name	Accuracy (%)
ResNet50 [18]	0.88
Sequential [19]	0.92
CNN [20]	0.93
DenseNet121 [21]	0.93
InceptionV3 [22]	0.94
Proposed Model(EfficientNetB4)	0.95

A comparison of different deep learning models based on their accuracy for a particular task is shown in Table II. ResNet50, Sequential CNN, Dense Net 121, Inception V3, and the suggested EfficientNetModel are all included in the comparison. Among these, Inception V3 attains maximum accuracy at 0.94 while the ResNet50 model has the lowest accuracy at 0.88. With an accuracy of 0.95, the suggested model making use of EfficientNet using CBAM Attention and Monte Carlo Dropout for uncertainty estimation beats all the other models, demonstrating its superior performance on the task at hand.

V. RESULTS AND DISCUSSION

The proposed Efficient NetB4 model demonstrates an accuracy of 95 % and strong melanoma classification performance. The precision and recall are (0.98, 0.91) for melanoma and (0.92, 0.98) for not melanoma classes. From the confusion matrix, it is clear that the misclassification rate is low thereby confirming reliable discrimination between classes. The comparative study shows that proposed approach outperforms CNN based methods, showing correct classification with low uncertainty thereby validating its effectiveness for melanoma detection.

VI. CONCLUSION

A new deep learning framework for melanoma recognition is proposed in this study that includes the CBAM attention mechanism, focal loss and Monte Carlo Dropout (MC Dropout) in order to prove uncertainty estimation. The integration has significantly improved the performance and dependability. These advanced techniques allowed our model to concentrate on key characteristics and address the problem of class imbalance in order to produce accurate forests with uncertainty. The proposed model made use of CBAM attention and the Monte Carlo dropout method for uncertainty estimation and the accuracy of the model is 95, and AUC is 0.98, which indicates it has a lot of potential for actual clinical use in the future. Our approach is novel in that it simultaneously optimizes the attention mechanism, focus loss and uncertainty estimation. Admittedly, in the presence of predictions and impossibilities, perhaps it is indeed that the model offers some kind of practical way to aid early diagnosis of skin cancer. The study is clinically relevant to image classification-based clinical applications, and it highlights the advantage of applying cutting-edge AI techniques for improved decision diagnosis and support accuracy. Future work can incorporate multi-class lesion classification to further improve robustness.

REFERENCES

[1] Shukla, Man Mohan, B. K. Tripathi, Tanay Dwivedi, Ashish Tripathi, M. M. Shukla, B. K. Tripathi, T. Dwivedi, A. Tripathi, and B. K. Chaurasia, "A hybrid CNN with transfer learning for skin cancer disease detection," *Med. Biol. Eng. Comput.*, vol. 62, no. 10, pp. 3057–3071, 2024.

[2] H. Naseri and A. A. Safaei, "Diagnosis and prognosis of melanoma from dermoscopy images using machine learning and deep learning: a systematic literature review," *BMC Cancer*, vol. 25, no. 1, p. 75, 2025.

[3] G. Yang, S. Luo, and P. Greer, "Advancements in skin cancer classification: a review of machine learning techniques in clinical image analysis," *Multimedia Tools Appl.*, vol. 84, no. 11, pp. 9837–9864, 2025.

[4] A. Patil, A. Mehto, and S. Nalband, "Enhancing skin lesion diagnosis with data augmentation techniques: a review of the state-of-the-art," *Multimedia Tools Appl.*, pp. 1–40, 2024.

[5] B. Lambert, F. Forbes, S. Doyle, H. Dehaene, and M. Dojat, "Trustworthy clinical AI solutions: A unified review of uncertainty quantification in deep learning models for medical image analysis," *Artif. Intell. Med.*, vol. 150, p. 102830, 2024.

[6] S. Agius, C. Magri, and V. Cassar, "A cognitive task analysis for developing a clinical decision support system for emergency triage," *J. Emerg. Nurs.*, 2025.

[7] V. Singh, K. A. Sultanpure, and H. Patil, "Frontier machine learning techniques for melanoma skin cancer identification and categorization: an in-depth review," *Oral Oncol. Rep.*, vol. 9, p. 100217, 2024.

[8] P. Zhang and D. Chaudhary, “Hybrid deep learning framework for enhanced melanoma detection,” *IEEE Trans. Comput. Biol. Bioinf.*, 2025.

[9] S. Nazari and R. Garcia, “Going smaller: Attention-based models for automated melanoma diagnosis,” *Comput. Biol. Med.*, vol. 185, p. 109492, 2025.

[10] P. K. Veni and A. Gupta, “Revolutionizing acne diagnosis with hybrid deep learning model integrating CBAM and capsule network,” *IEEE Access*, vol. 12, pp. 82867–82879, 2024.

[11] J. Chen *et al.*, “Pigmented skin disease classification via deep learning with an attention mechanism,” *Appl. Soft Comput.*, vol. 170, p. 112571, 2025.

[12] G. Doğan, “Performance analysis of attention mechanisms for melanoma cancer detection,” in *Proc. 8th Int. Artif. Intell. Data Process. Symp. (IDAP)*, 2024, pp. 1–6.

[13] M. J. Abbas *et al.*, “C3BAM-XAI: Convolutional block attention module enhanced explainable artificial intelligence-based Parkinson’s disease stage classification,” *Cogn. Comput.*, vol. 17, no. 3, p. 111, 2025.

[14] B. Mittal, “Melanoma cancer image dataset,” Kaggle, 2023. [Online]. Available: https://www.kaggle.com/datasets/bhaveshmittal/melanoma-cancer-dataset. [Accessed: Aug. 23, 2025].

[15] R. Agrawal, N. Gupta, and A. S. Jalal, “CACBL-Net: A lightweight skin cancer detection system for portable diagnostic devices using deep learning–based channel attention and adaptive class balanced focal loss function,” *Multimedia Tools Appl.*, pp. 1–24, 2024.

[16] B. Mittal, “Melanoma cancer image dataset,” *Kaggle*, 2023. [Online]. Available: https://www.kaggle.com/datasets/bhaveshmittal/melanoma-cancer-dataset. [Accessed: Aug. 23, 2025].

[17] Z. Ji *et al.*, “Efam-net: A multi-class skin lesion classification model utilizing enhanced feature fusion and attention mechanisms,” *IEEE Access*, 2024.

[18] M. K. Ambar, H. Oztoprak, and K. Yurtkan, “Optimizing ResNet-50 for multiclass classification: A multi-stage learning approach,” *IEEE Access*, 2025.

[19] Y. Jia, L. Dong, and Y. Jiao, “Medical image classification based on contour processing attention mechanism,” *Comput. Biol. Med.*, vol. 191, p. 110102, 2025.

[20] A. Siddique, K. Shaukat, and T. Jan, “An intelligent mechanism to detect multi-factor skin cancer,” *Diagnostics*, vol. 14, no. 13, p. 1359, 2024.

[21] M. Combalia, N. C. F. Codella, V. Rotemberg, B. Helba, V. Vilaplana, O. Reiter, C. Carrera, A. Barreiro, A. C. Halpern, S. Puig, and J. Malvehy, “BCN20000: Dermoscopic lesions in the wild,” *arXiv preprint* arXiv:1908.02288, 2019.

[22] N. C. F. Codella, D. Gutman, M. E. Celebi, B. Helba, M. A. Marchetti, S. W. Dusza, A. Kalloo, K. Liopyris, N. Mishra, H. Kittler, and A. Halpern, “Skin lesion analysis toward melanoma detection: A challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), hosted by the International Skin Imaging Collaboration (ISIC),” in *Proc. IEEE 15th Int. Symp. Biomed. Imag. (ISBI)*, Apr. 2018, pp. 168–172, doi: 10.1109/ISBI.2018.8363547.

[23] P. Tschandl, C. Rosendahl, and H. Kittler, “The HAM10000 dataset: A large collection of multi-source dermatoscopic images of common pigmented skin lesions,” *Sci. Data*, vol. 5, no. 1, Art. no. 180161, Aug. 2018, doi: 10.1038/sdata.2018.161.