Comparative Analysis of Convolutional Neural Networks and Transfer Learning for Skin Cancer Classification

[1]Luiz Filipe de Lima Gomes Santos, [2]Lucy Mari Tabuti, [3]Dheiver Francisco Santos

[1]XP Educação – IGTI (XPE) – Maceió, AL – Brazil

[2]POLI-USP – Universidade de São Paulo (USP) – São Paulo – SP – Brasil

[3]Cognai Healthtech – Maceió – AL – Brazil

luizfilipelgs@gmail.com, lucymari@gmail.com, dheiver.santos@gmail.com

Abstract. *Skin cancer is one of the most common types of cancer, and early detection is critical for effective treatment. This work compares three deep learning models for melanoma classification: a simple CNN, a CNN with Dropout, and MobileNetV2 with transfer learning. All were trained on a public dataset of 10,000 images resized to 64×64 pixels. While this resolution improved training speed, it likely hindered MobileNetV2's performance due to architectural constraints. We evaluated accuracy, precision, recall, and F1-score. Results show that lightweight CNN outperformed MobileNetV2, emphasizing the need to align input resolution with model architecture.*

1. Introduction

Skin cancer is one of the most common and potentially lethal cancers worldwide. Among its types, melanoma stands out due to its high mortality rate when not diagnosed early. According to the [World Health Organization 2025], early detection significantly improves patient survival rates and reduces treatment costs. Traditional diagnosis methods, based on clinical observation and histopathology, are effective but often time-consuming and subject to human variability.

In recent years, Convolutional Neural Networks (CNNs) have shown great promise in medical image analysis, including skin lesion classification tasks [Santos 2022], [Litjens et al. 2017], [Esteva et al. 2017]. Their ability to extract and learn complex visual features has led to significant improvements in diagnostic accuracy, sometimes surpassing human specialists [Haenssle et al. 2018].

Previous studies using CNNs on ISIC datasets with 64×64 resolution images have already shown promising accuracy levels around 89% [Santos 2022]. Inspired by such results, this study explores alternative architectures and transfer learning approaches to improve performance under the same resolution constraint.

This study proposes a comparative analysis of three deep learning architectures applied to binary classification of skin cancer images (melanoma vs. non-melanoma): (i) a simple CNN, (ii) a CNN with Dropout regularization, and (iii) MobileNetV2 with transfer learning. All models were trained using the public "Melanoma Skin Cancer Dataset of 10,000 Images" [Javid 2022], resized to a resolution of 64×64 pixels.

The choice of this reduced input size aimed at accelerating training and reduce computational demands. However, this decision may have compromised the performance of MobileNetV2, a model originally evaluated using input resolutions ranging from 96×96 to 224×224 pixels [Sandler et al. 2018]. As discussed later in this paper, this architectural mismatch potentially impacted on its classification ability.

The goal of this work is to analyze the performance trade-offs between lightweight CNNs and more complex transfer learning models under resolution constraints, contributing to the development of fast, accurate, and accessible diagnostic tools.

2. Methodology

2.1. Dataset and Preprocessing

We used the public *Melanoma Skin Cancer Dataset of 10,000 Images* [6], which contains thermoscopic images labeled as either benign or malignant. The dataset was already organized into separate folders for training and testing. Specifically, the training set included 5000 benign and 4605 malignant images, while the test set included 500 images from each class.

All images were resized to 64×64 pixels and normalized to the [0, 1] range. The decision to use a reduced resolution was made to optimize training speed and computational resource usage, which is particularly important for lightweight models. However, this choice may have limited the performance of deeper networks such as MobileNetV2, which was originally evaluated with input resolutions ranging from 96×96 to 224×224 pixels [Sandler et al. 2018].

2.2. Data Augmentation

To improve generalization and reduce overfitting, data augmentation techniques were applied during training using Keras' ImageDataGenerator. The transformations included random horizontal and vertical flips, slight zoom, and rotation. These augmentations aimed to simulate real-world variability in skin lesion images, improving robustness and model performance.

2.3. Training Setup

The training and test datasets were kept as originally provided by the dataset author, with no additional splitting or reshuffling. All models were implemented using the Keras API and trained using the binary crossentropy loss function and the Adam optimizer. The training process was configured with a batch size of 32 and allowed to run for up to 20 epochs. To prevent overfitting, the EarlyStopping callback was employed, monitoring the validation loss with a patience of 3 epochs.

No explicit validation set was defined; instead, model performance was evaluated using the test set after training. The training history, including accuracy and loss over the epochs, was recorded for performance analysis and comparative evaluation between models.

2.4. Model Architectures

We trained and compared three deep learning models:

I. Model A: Simple CNN

A baseline architecture consisting of two convolutional layers with ReLU activation, each followed by max pooling. The output is flattened and passed through a fully connected layer with 64 units and ReLU activation. The final output layer uses a sigmoid activation function to perform binary classification.

```
model = Sequential()
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=input_shape))
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D((2, 2)))
model.add(Flatten())
model.add(Dense(64, activation='relu'))
model.add(Dense(1, activation='sigmoid'))
```

Figure 1. Architecture of Model A – Simple CNN

II. Model B: CNN with Dropout

An extended version of Model A, with an additional convolutional block and a denser fully connected layer. A dropout layer was added after the dense layer to reduce overfitting and improve generalization by randomly disabling neurons during training.

```
model = Sequential()
model.add(Conv2D(32, (3, 3), activation='relu', input_shape=input_shape))
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(64, (3, 3), activation='relu'))
model.add(MaxPooling2D((2, 2)))
model.add(Conv2D(128, (3, 3), activation='relu'))
model.add(MaxPooling2D((2, 2)))
model.add(Flatten())
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))
```

Figure 2: Architecture of Model B – CNN with Dropout.

III. Model C: MobileNetV2 (Transfer Learning)

MobileNetV2 is a pre-trained convolutional network originally designed for efficient classification on mobile devices [Sandler et al. 2018]. In this model, we used MobileNetV2 as a frozen feature extractor by setting its convolutional base as non-trainable. The top layers were replaced with a global average pooling layer, followed by a dropout layer and a final dense layer with sigmoid activation for binary classification. All input images were resized to 64×64 pixels to match the previous models, although MobileNetV2 is typically optimized for higher resolutions. This architectural mismatch may have contributed to the lower performance observed in this model.

```
base_model = MobileNetV2(input_shape=input_shape, include_top=False, weights='imagenet')
base_model.trainable = False

x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dropout(0.5)(x)
output = Dense(1, activation='sigmoid')(x)

model = Model(inputs=base_model.input, outputs=output)
```

Figure 3: Architecture of Model C – MobileNetV2 (Transfer Learning).

3. Experimental Results

This section presents the quantitative and visual results obtained from the training and evaluation of the three proposed models. The goal is to assess the classification performance of each architecture and compare their generalization capabilities under the same input resolution constraint (64×64 pixels). The evaluation was based on standard classification metrics and visual inspection of learning curves, confusion matrices, and prediction examples.

3.1. Validation Curves

The validation accuracy and loss curves over the training epochs are shown below for the three models. Model B (CNN with Dropout) demonstrated more stable training behavior and better generalization, while Model C (MobileNetV2) suffered from performance degradation, possibly due to its architectural reliance on higher-resolution inputs.

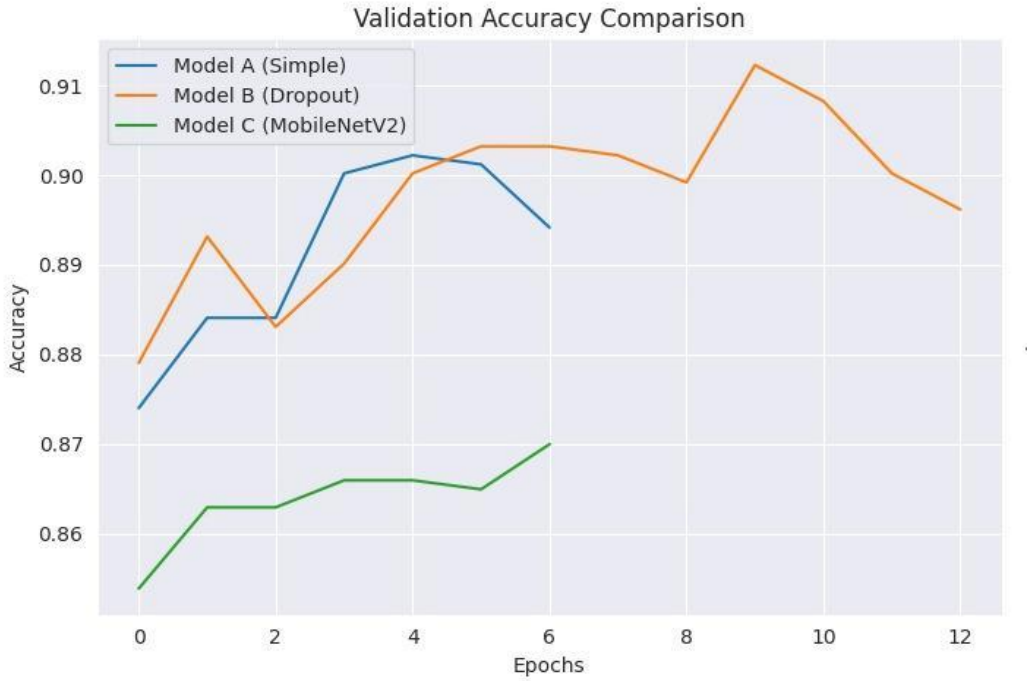

Figure 4. Validation accuracy for Models A, B, and C.

Figure 5. Validation loss curves for Models A, B, and C.

3.2. Final Evaluation Metrics

To quantify the classification performance, we calculated accuracy, precision, recall, and F1-score for each model using the test set. The results are summarized in the table below.

Table 1. Evaluation metrics for melanoma class on the test set.

	Accuracy	Precision	Recall	F1-score
Model A	0.900	0.912	0.886	0.899
Model B	0.912	0.931	0.890	0.910
Model C	0.866	0.896	0.828	0.861

Model B achieved the highest scores across all metrics, suggesting that the addition of dropout and a deeper convolutional structure helped improve generalization. In contrast, Model C underperformed despite leveraging transfer learning, likely due to its incompatibility with the reduced input size.

3.3. Confusion Matrices

Confusion matrices were generated to visualize the classification performance per class. Model B, showed in Figure 7, achieved the best balance between sensitivity and specificity, with fewer false negatives compared to Models A and C, like showed in Figure 6 and Figure 8.

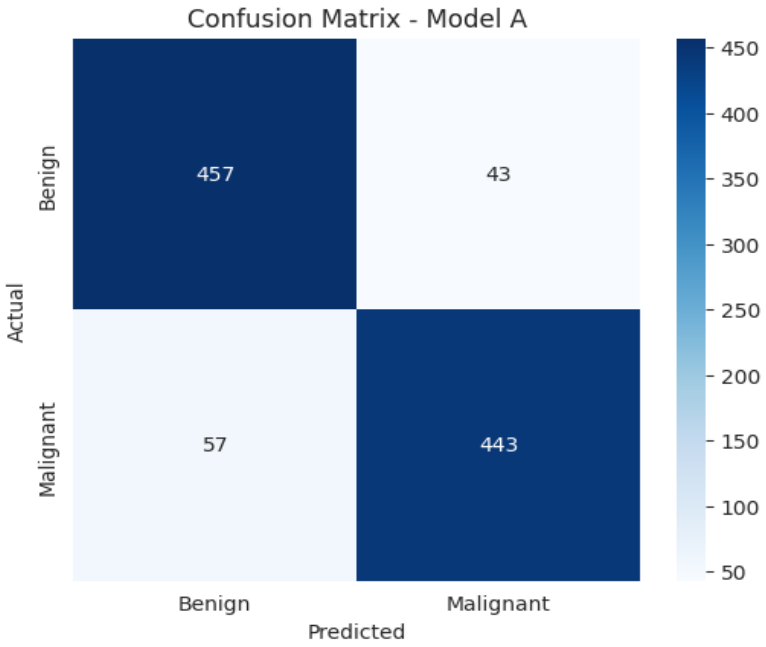

Figure 6. Confusion matrix for Model A.

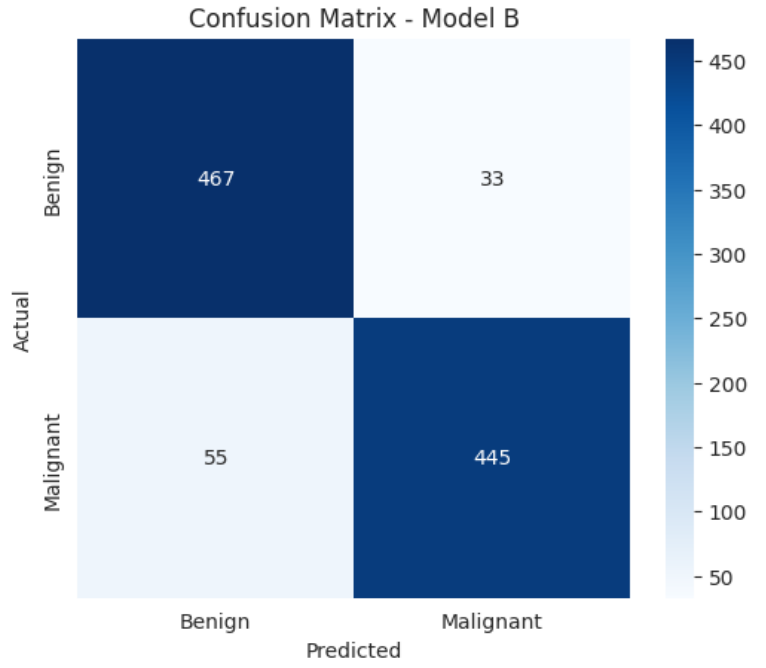

Figure 7. Confusion matrix for Model B.

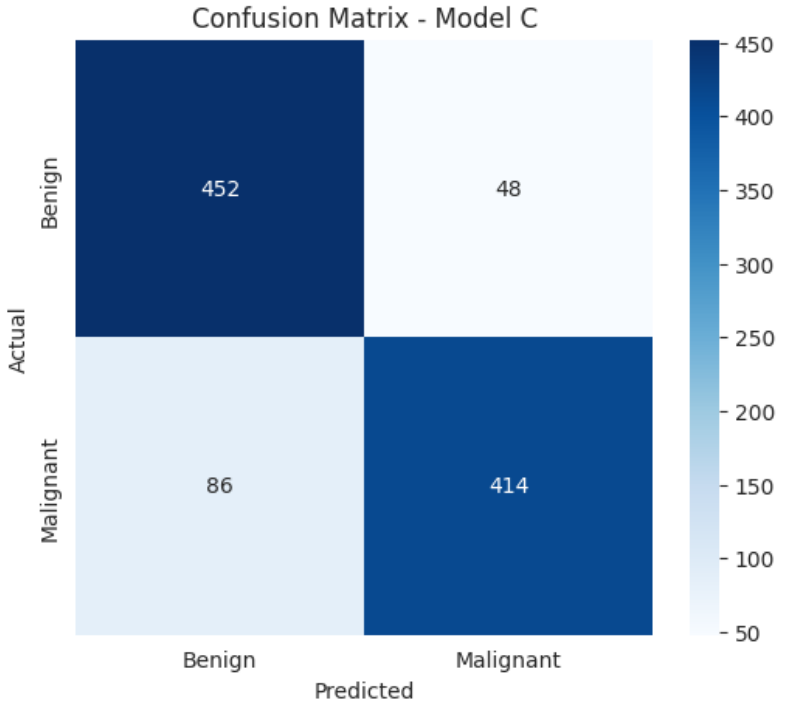

Figure 8. Confusion matrix for Model C.

3.4. Predictions on Test Images

To qualitatively assess model behavior, Figure 9 displays predictions from each model (columns) on the same set of test images (rows). Each row represents a single skin lesion (one benign and one malignant), and each column corresponds to a different model. The predicted class and associated confidence score are shown above each image. Correct predictions are displayed in blue, while incorrect predictions appear in red, allowing a clear visual comparison of the models' performance on the same inputs.

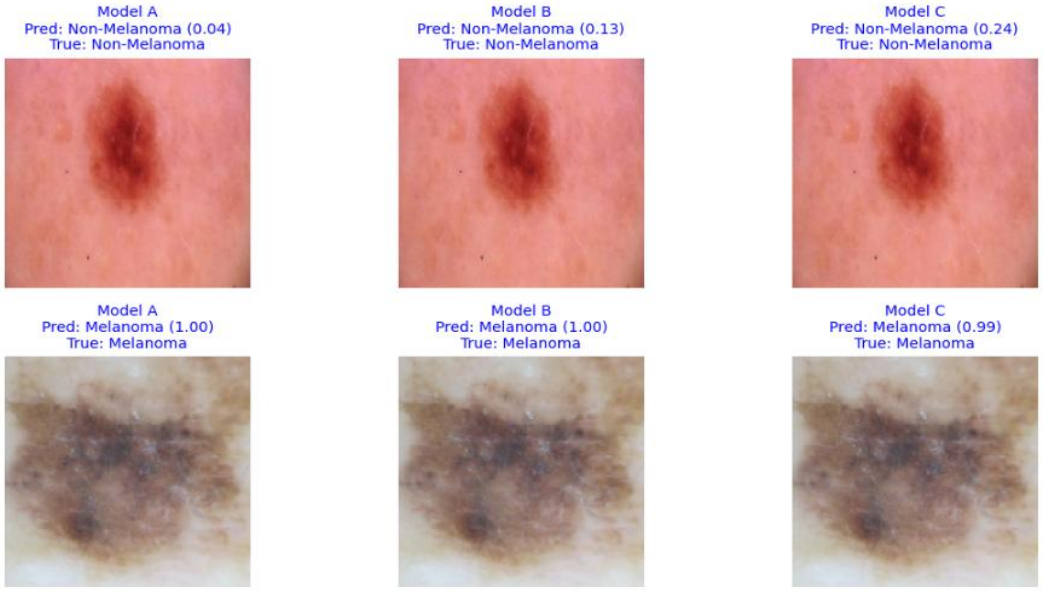

Figure 9. Predictions on test images by Model A, B and C.

4. Discussion

The results demonstrate that architectural simplicity, when well-regularized, can outperform more complex models under resolution constraints. Model B (CNN with Dropout) achieved the best overall performance, with 91.2% accuracy and the highest precision, recall, and F1-score, likely due to deeper feature extraction and reduced overfitting.

Model A, despite its simpler structure, showed competitive performance (90.0% accuracy), reinforcing the effectiveness of lightweight CNNs for binary image classification tasks.

In contrast, Model C (MobileNetV2) underperformed (86.6% accuracy), possibly due to its reliance on higher-resolution inputs. The reduced input size (64×64) may have limited its ability to extract meaningful features, despite using transfer learning.

Visual predictions support these findings: Model C had more misclassifications, including confident errors. Overall, this study suggests that tailored CNNs may be more effective than pre-trained architectures when computational constraints and image resolution are limiting factors.

5. Conclusion and Future Work

This study compared three deep learning models for melanoma classification using dermoscopic images resized to 64×64 pixels. Model B (CNN with Dropout) achieved the best performance across all metrics, confirming that architectural depth combined with regularization can significantly enhance generalization.

Although MobileNetV2 is a powerful pre-trained model, its performance was limited by the reduced input resolution, highlighting the importance of aligning preprocessing choices with model design.

Future work may explore higher-resolution inputs, fine-tuning of transfer learning models, and model deployment in real-world scenarios, such as mobile health applications or clinical decision support systems.

7. References

[World Health Organization 2025] World Health Organization. Radiation: Ultraviolet (UV) radiation and skin cancer. Available at: https://www.who.int/news-room/questions-and-answers/item/radiation-ultraviolet-(uv)-radiation-and-skin-cancer. Accessed: March 2025.

[Santos 2022] Santos, D. (2022). *Melanoma Skin Cancer Detection using Deep Learning*. medRxiv. https://doi.org/10.1101/2022.02.02.22269505

[Litjens et al. 2017] Litjens, G., Kooi, T., Bejnordi, B. E., Setio, A. A. A., Ciompi, F., Ghafoorian, M., et al. (2017). A survey on deep learning in medical image analysis. Medical Image Analysis, 42, 60–88. https://doi.org/10.1016/j.media.2017.07.005

[Esteva et al. 2017] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M., & Thrun, S. (2017). Dermatologist-level classification of skin cancer with deep neural networks. Nature, 542(7639), 115–118. https://doi.org/10.1038/nature21056

[Haenssle et al. 2018] Haenssle, H. A., Fink, C., Schneiderbauer, R., Toberer, F., Buhl, T., Blum, A., et al. (2018). Man against machine: diagnostic performance of a deep learning convolutional neural network for dermoscopic melanoma recognition in comparison to 58 dermatologists. Annals of Oncology, 29(8), 1836–1842. https://doi.org/10.1093/annonc/mdy166

[Javid 2022] Javid, M. H. (2022). Melanoma Skin Cancer Dataset of 10,000 Images. Kaggle. https://doi.org/10.34740/KAGGLE/DSV/3376422

[Sandler et al. 2018] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., & Chen, L. C. (2018). MobileNetV2: Inverted Residuals and Linear Bottlenecks. arXiv:1801.04381. https://arxiv.org/abs/1801.04381