








SkinSight: A Melanoma Detection App Based on Deep Learning Models with On-Device Inference

Adrian Chavez-Ramirez¹ , Andrea Romero-Ramos¹ ,
Monica Aguirre-Ortega¹ , Samantha Aguilar-Gameros¹ ,
and Graciela Ramirez-Alonso² 

¹ Facultad de Medicina y Ciencias Biomédicas, Universidad Autónoma de Chihuahua, 31125 Chihuahua, Chihuahua, Mexico

² Facultad de Ingeniería, Universidad Autónoma de Chihuahua, 31125 Chihuahua, Chihuahua, Mexico
galonso@uach.mx

Abstract. A timely and accurate skin cancer diagnosis is a key factor in reducing mortality rates, especially with melanoma which often resembles in its early stages with moles. Convolutional neural networks (CNNs) are models commonly used to classify dermoscopy images into benign or malignant. CNNs are frequently implemented on Graphical Processing Units (GPUs), which are not always available in rural areas. This paper compares three CNNs to classify benign and malignant melanoma images. We select the most appropriate neural architecture by comparing accuracy results and model lightness to load it on a mobile device. With this strategy, the training of the CNN is performed on the GPU and the inference in portable devices that can be used in rural areas. The developed app is named SkinSight. This app was evaluated with images of two different datasets achieving competitive results compared to state-of-the-art models. Considering that most people have a mobile device, this app could be used in areas where it is difficult to have specialized GPUs and highly trained personnel in cancer detection.

Keywords: melanoma detection · convolutional neural network · mobile application

1 Introduction

According to the American Cancer Society, melanoma is the most dangerous type of skin cancer; its early diagnosis is essential for successful treatment and patient survival [1]. In a study published in the Skin Cancer Foundation [3], late diagnosis of melanoma is a significant problem in many parts of the world, including Latin America, where lack of access to health services and awareness about skin cancer contributes to a late diagnosis. In 2020, according to data from the International Agency for Research in Cancer of the World Health Organization, through the

GLOBOCAN project, the incidence of Melanoma in Mexico was 2,051 cases with 773 deaths [4].

The diagnosis of melanomas is mainly made by visually inspecting skin lesions by highly trained dermatologists. Asymmetry, border, color, diameter, and lesion enlargement are the standard features that specialists consider. Another common way to diagnose cancer is by performing a biopsy, a pathological examination that takes much time and resources to provide the results. The Sierra Tarahumara is a mountain range part of the Sierra Madre Occidental located in Chihuahua. This rural and remote area lacks sufficient pathologists and medical resources to diagnose and treat skin cancer. The lack of information and awareness about this type of cancer in these communities can lead to delayed seeking care and late diagnosis of the disease. This problem can seriously affect the population's health and lead to higher mortality and morbidity rates in the region. A comprehensive approach is needed to address the lack of access to pathology services, including actions to increase skin cancer awareness, improve local doctors' training, and provide resources and technology for diagnostic testing and treatment.

Deep learning techniques, especially convolutional neural networks (CNNs), have been widely used in different image recognition tasks to automatically classify specific patterns on images [11]. Particularly for classifying skin cancer, different CNN models have been proposed achieving very accurate classification results [7, 16, 22].

Unfortunately, these systems have not yet been incorporated into daily clinical practice because most CNN models need the usage of Graphical Processing Units (GPUs), a hardware not very common in most hospitals. As an alternative to using expensive hardware equipment, TensorFlow (an open-source machine learning ML framework) has launched a lightweight version named TensorFlow Lite (TFLite) [5]. TFLite is optimized for deploying deep learning models on mobile and embedded devices with limited computational resources. Then, CNNs can be implemented in low-cost, low-power, portable, easy-to-use devices for classification and detection tasks. The training is performed on the GPU, but the inference can be executed on mobile devices, also known as on-device inference.

This work presents a comparison of state-of-the-art CNN models to classify images into benign or malignant melanoma lesions automatically. These models are trained and tested on two skin cancer datasets, demonstrating their robustness in different scenarios. The inference of the selected CNN model can be performed on a mobile device, known as on-device inference. The TFLite framework, in combination with Android Studio, allows us to convert the CNN model to a light version capable of working on low-cost, low-power devices. In this way, this CNN can be easily used by medical specialists with access to dermoscopy images and have the opportunity to diagnose suspicious cases in an early manner. Even when this methodology has already been implemented in recent research, most of them only evaluate their proposal in one dataset with few samples, achieve low-performance results, or perform the inference of the model in a server computer. In our proposal, we could maintain a balance

between accurate performance results considering two different datasets demonstrating the robustness of our proposal. We named our application SkinSight, which can be loaded on Android devices. Considering that most people have a smartphone, this tool could be used where it is difficult to have highly specialized GPUs and/or trained personnel in cancer detection. It is worth mentioning that this paper aims to identify the best CNN configuration that achieves a comparable performance with state-of-the-art models trained and tested on GPUs and with those developed to be used in portable devices.

2 Literature Review

The International Skin Imaging Collaboration (ISIC) is a global organization with an online repository of dermoscopic and clinical images of skin lesions [2]. The objective is that researchers from all over the world can work in the development of computer-aided systems to detect and diagnose melanoma and other skin cancers. With the advancement in computer vision algorithms based on deep learning models, different researchers have reported accurate results in classifying benign and malignant skin lesions. Cassidy et al. performed a benchmark study in [9] with images of the ISIC dataset and 19 state-of-the-art deep learning architectures. The VGG19, DenseNet121, and EfficientNetB2 architectures achieved the best area under the Receiver Operating Characteristic Curve (AUC) results. Benyahia, Meftah, and Lezoray [8] also investigate the efficiency of 17 deep learning architectures and 24 machine learning classifiers using the ISIC dataset. They concluded that the DenseNet201 neural architecture combined with the Cubic SVM algorithm produces the best classification results.

Rehman et al. [25] use a modified pre-trained DenseNet201 by staking three convolutional layers at the end of the model, followed by a global average pooling, a batch normalization, and two dense layers. The authors used a contrast stretching enhanced technique to improve the quality of the images reporting an average accuracy of 95.5%. In [21], was adapted a ResNet101 architecture to classify benign and malignant skin cancer images. Two convolutional layers were included at the end of the model, followed by pooling and two fully connected layers. The authors reported an average accuracy of 90.67%.

All these previous research papers perform their training and testing in a specialized GPU, achieving state-of-the-art performance in skin lesions classification tasks. After deeply analyzing their results, we select the ResNet101, DenseNet201, and a CNN of the EfficientNet family in our experiments. The accurate reported results and reduced number of parameters in these neural architectures make them ideal candidates for our research.

Figure 1 shows a block diagram of the process we follow in developing our SkinSight app. First, it is necessary to train the different deep learning models on TensorFlow with the appropriate datasets and compare their performance to select the most appropriate model. Then, convert the selected CNN to TensorFlow Lite. Next, set up Android Studio for Android App development with the appropriate Android SDK and NDK components installed, add TensorFlow Lite

dependencies, and copy the TF-Lite model into the project. The TFLite interpreter is necessary to load the model in the project. A user interface is designed to create the views and controls to interact with the model and display the prediction results appropriately. Then, connect an Android device to the computer and build the app with Android Studio. Finally, test SkinSight with images to confirm that the CNN model works as required.

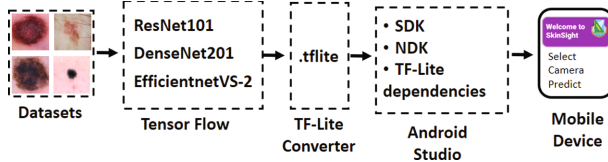


Fig. 1. Block diagram of deploying a CNN in a mobile device using TF-Lite and Android Studio.

The general methodology of performing the training of the CNN in the GPU and the inference in a mobile device (to be used by the medical sector) has already been proposed in different research papers. In [19] is presented a mobile app to classify skin diseases considering their severity based on the MobileNetV2 architecture. A dataset of 1,220 images is processed, achieving an accuracy of 94.32% in the classification task. In [14], a dataset of 2,358 images was classified as melanoma or benign using the InceptionV3 neural architecture. The accuracy reported by the authors is 81%. Dai et al. [10] presented an on-device inference app using 10,015 images. The accuracy achieved by the model was 75.2%. In [15] is presented an augmented reality app that classifies skin lesions for identifying melanoma. The app continuously tracks the lesion, implementing different image pre-processing algorithms to remove hair and segment the lesion before analyzing the image in the CNN model. Their method achieved an accuracy of 78.8%. Kousis et al. [20] load a light version of a DenseNet169 network on a mobile Android device to classify benign or malignant images. The DenseNet169 model achieved an accuracy of 91.10%, considering a dataset of 10,015. The authors mentioned that when testing their app in a real environment, it was necessary to transfer the image to a server for better performance. In [12], the MobileNetV2 architecture classifies skin lesion images considering three datasets. The overall accuracy performance reported when testing their proposal in a new dataset with the mobile app was 91.33%. Arani et al. [6] presented the Melanlysis app for detecting skin cancer based on the EfficientNetLite-0 architecture. The authors use only the dataset's dermoscopy images, achieving an accuracy of 94%. In [13] is presented a lesion segmentation and classification method based on a DenseNet201 model loaded on a mobile device. The classification task considers the identification of seven skin lesion classes achieving an accuracy of 89%.

3 Methodology

3.1 Deep Learning Models

The ResNet (Residual Neural Network) architecture introduces the concept of residual or skip connection to address the vanishing gradient problem present in deep neural networks [17]. The Residual Blocks of the ResNet model have convolutional and batch normalization layers and ReLu activation functions. The number of residual blocks defines the variant of the ResNet architecture. We select the ResNet101 in our experiments considering the results reported in [21].

DenseNet, or Dense Connected Convolutional Network, uses the concept of dense blocks to connect the output of every other layer within each of its blocks [18]. That is, the output of each layer is concatenated before passing it to the input of the subsequent layer within each dense block. To reduce the spatial dimensions between dense blocks and the number of channels, DenseNet defines Transition Layers. Similar to ResNet, DenseNet defines different variants, and in our experiments, the DenseNet201 is selected according to the results in [25].

EfficientNet is a family of deep neural network architectures that use a neural architecture search method to uniformly scale the network’s depth, width, and input image size. EfficientNetV2 [24] aims to optimize the training speed and parameter efficiency. Regularization techniques are adaptively adjusted during training, considering different input image sizes. The authors define this particularity as *Progressive Learning with Adaptive Regularization*. In TensorFlow are implemented seven versions of EfficientNetV2. In our experiments, we select the EfficientNetV2-S variant because it has almost the same number of parameters as DenseNet201.

In order to adapt these three different CNN architectures to the skin cancer datasets, we consider two options. The first one only includes a global average pooling in the last convolutional layer of these architectures, followed by a fully connected layer. Inspired in [25], a second option considers including three convolutional layers, a global average pooling, and a batch normalization, followed by fully connected with dropout layers. A transfer learning strategy was used to train these neural architectures where initially, only the extra layers were trained by ten epochs (freezing the layers of the CNN architectures). Then, a fine tune strategy unfreezes 20% of the CNN architecture, and a new training is performed with a reduced learning rate.

3.2 TensorFlow Lite (TFLite)

TensorFlow Lite (TFLite) [5] is a lightweight deep learning framework specifically designed for deploying CNN models to mobile and embedded devices created by Google. TFLite optimizes the size and speed of the models without neglecting their performance. TFLite uses quantization methods to compress the deep learning model by using fewer bits to represent model parameters [23].

Once the model is converted to a TFLite format, the integrated development environment (IDE) of Android Studio for Android App is used to load the CNN model into the mobile device. The TFLite interpreter is in charge of running the inference of the model and producing the predictions. Then, deploying deep learning models on mobile devices is possible by combining Tensor Flow, TFLite, and Android Studio.

4 Experimental Settings and Results

In our experiments, we use two datasets presented on Kaggle that consider images of the ISIC challenges. Dataset one (DS1)¹ has 3,297 dermoscopic images. 1,800 images are classified as benign and 1,497 as malignant, respectively. Kaggle provides a data partition where 80% of the data is separated to train and 20% to test. In our experiments, the training data was re-partitioned into train and validation with a final distribution of 60% to train, 20% to validate, and 20% to test. The second dataset (DS2)² has 10,605 images. Kaggle defines 9,605 for training and 1,000 for testing. Same as the previous dataset, the training data was re-partitioned to provide a validation set. The final data split corresponds to 80% to train, 10% to validate, and 10% to test.

The training of the CNN models used in this work is performed on Google Colaboratory, a cloud-based platform with pre-installed libraries and dependencies. In our case, we use the TensorFlow library to train the CNN models. Table 1 shows the accuracy classification results of the different CNN architectures. The second column specifies if the CNN considers the three convolutional extra layers, global average pooling, and batch normalization, followed by fully connected and dropout layers. The third column indicates the number of parameters of each CNN. The fourth and fifth columns indicate the accuracy percentage achieved by each CNN.

Table 1. Accuracy results of the different neural architectures.

CNN model	Extra layers	Parameters	DS1	DS2
ResNet101	No	42,660,225	87.87%	92.10%
ResNet101	Yes	43,575,129	86.97%	92.40%
DenseNet201	No	18,832,905	86.06%	91.9%
DenseNet201	Yes	19,189,785	86.96%	90.7%
EfficientnetV2-S	No	20,332,641	85.61%	91.40%
EfficientnetV2-S	Yes	20,953,401	87.57%	91.90%

¹ <https://www.kaggle.com/datasets/fanconic/skin-cancer-malignant-vs-benign>.

² <https://www.kaggle.com/datasets/hasnainjaved/melanoma-skin-cancer-dataset-of-10000-images>.

The accuracy results of the models are very similar. The best accuracy and the model with fewer parameters are highlighted in bold. ResNet101 obtains the best classification results but is the CNN with the largest number of parameters. EfficientnetV2-S and DenseNet201 obtain comparable performance, but in our implementation, it is very important to have a reduced number of parameters because our objective is to deploy the CNN model in an Android application running on a mobile device. For this reason, we select the DenseNet201 model. Figure 2 shows the confusion matrix results obtained with the DenseNet201 model considering the two datasets.

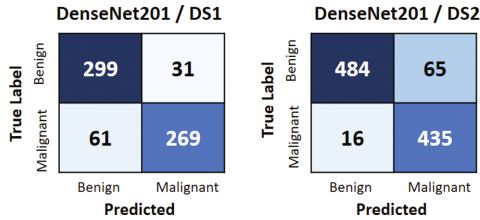


Fig. 2. Confusion matrix results

By visually inspecting the images of the datasets, we realize that some of them are very difficult to classify as benign or malignant. Figure 3 shows some of them where, despite being difficult samples, the DenseNet201 model correctly classifies them.

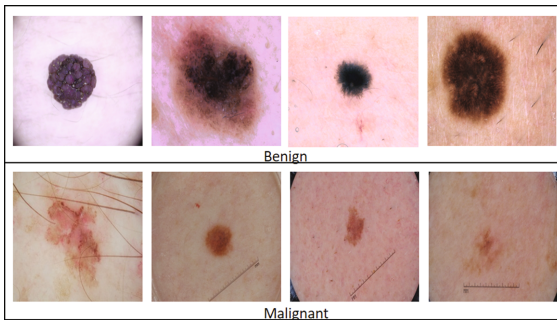


Fig. 3. Examples of difficult samples of the datasets

Once the model was trained, it was converted to a light version with TF-Lite and loaded into the mobile device using Android Studio for Android App. Figure 4 shows the final user interface designed for SkinSight with prediction results. SkinSight can load images from the smartphone gallery. With this option, we could select the testing images of DS1 and DS2 and confirm that the accuracy

performance of the model is maintained on the light version obtaining the same results reported on the confusion matrix of Fig. 2. By comparing these results with those models reported in Sect. 2, our accuracy performance is superior to most mobile apps. Only two of them achieved better results. The first only considers one dataset of few samples (1,220 images), and the second eliminates images not obtained with a dermoscopy (the ISIC dataset has images obtained with simple cameras and are commonly incorrectly classified).

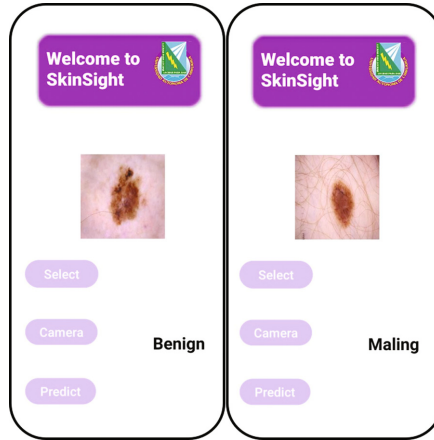


Fig. 4. Prediction results of the SkinSight app.

5 Conclusions

This paper presents the process we follow to design an Android app named SkinSight to detect melanoma automatically. First, we compare the performance of state-of-the-art CNN models trained and tested with images of two datasets of the ISIC challenge. The accuracy results obtained with EfficientnetV2-S, ResNet101, and DenseNet201 are very similar. However, considering that our objective is to develop a mobile app that medical personnel can use to diagnose suspicious cases early, we select the CNN model with the fewest parameters. The combination of using TensorFlow, TensorFlow Lite, and Android Studio offers a powerful solution for deploying deep learning models on mobile devices.

Recent models that surpass the results reported in this paper implement highly cost pre-processing techniques to remove noise and artifacts from the images. Also, some of these publications stack more than five machine learning algorithms, but the improvement is only 3% compared to our implementations. Considering that our SkinSight app is designed to be used by the medical sector with limited resources, we bear in mind a balance between accurate classification results and a few parameters of the model. In this paper, we only perform the

testing of SkinSight with images already analyzed by specialists. Because we want to bring this tool closer to rural areas of our location, our next step is to work with local medical doctors and patients already diagnosed with this disease and test the app in a real environment to identify how to handle different skin tonalities and factors not considered on the ISIC dataset.

References

1. Estadísticas importantes sobre el cáncer de piel tipo melanoma. <https://www.cancer.org/es/cancer/tipos/cancer-de-piel-tipo-melanoma/acerca/estadisticas-clave.html>. Accessed 13 Jun 2023
2. ISIC archive. <https://www.isic-archive.com/#!/topWithHeader/wideContentTop/main>
3. Skin cancer foundation. <https://www.skincancer.org/skin-cancer-information/melanoma/>. Accessed 13 Jun 2023
4. Skin cancer foundation. <https://gco.iarc.fr/today/fact-sheets-populations>. Accessed 20 Jun 2023
5. TensorFlow Lite. <https://www.tensorflow.org/lite?hl=es-419>. Accessed 01 Jun 2023
6. Arani, S., Zhang, Y., Rahman, M., Yang, H.: Melanlysis: a mobile deep learning approach for early detection of skin cancer. In: 2022 IEEE 28th International Conference on Parallel and Distributed Systems (ICPADS), Los Alamitos, CA, USA, January 2022, pp. 89–97. IEEE Computer Society (2022). <https://doi.org/10.1109/ICPADS56603.2022.00020>
7. Bansal, P., Garg, R., Soni, P.: Detection of melanoma in dermoscopic images by integrating features extracted using handcrafted and deep learning models. *Comput. Ind. Eng.* **168**, 108060 (2022). <https://doi.org/10.1016/j.cie.2022.108060>. <https://www.sciencedirect.com/science/article/pii/S0360835222001309>
8. Benyahia, S., Meftah, B., Lezoray, O.: Multi-features extraction based on deep learning for skin lesion classification. *Tissue Cell* **74**, 101701 (2022). <https://doi.org/10.1016/j.tice.2021.101701>
9. Cassidy, B., Kendrick, C., Brodzicki, A., Jaworek-Korjakowska, J., Yap, M.H.: Analysis of the ISIC image datasets: usage, benchmarks and recommendations. *Med. Image Anal.* **75**, 102305 (2022). <https://doi.org/10.1016/j.media.2021.102305>
10. Dai, X., Spasić, I., Meyer, B., Chapman, S., Andres, F.: Machine learning on mobile: an on-device inference app for skin cancer detection. In: 2019 Fourth International Conference on Fog and Mobile Edge Computing (FMEC), pp. 301–305 (2019). <https://doi.org/10.1109/FMEC.2019.8795362>
11. Deng, T.: A survey of convolutional neural networks for image classification: models and datasets. In: 2022 International Conference on Big Data, Information and Computer Network (BDICN), pp. 746–749 (2022). <https://doi.org/10.1109/BDICN55575.2022.00145>
12. Ech-Cherif, A., Misbhauddin, M., Ech-Cherif, M.: Deep neural network based mobile dermoscopy application for triaging skin cancer detection. In: 2019 2nd International Conference on Computer Applications and Information Security (ICCAIS), pp. 1–6 (2019). <https://doi.org/10.1109/CAIS.2019.8769517>
13. Emam Ananna, M., Nayeem, J., Jahangir Alam, M., Islam, S.: Skin cancer detection using machine learning framework with mobile application. In: 2023 7th International Conference on Trends in Electronics and Informatics (ICOEI), pp. 1073–1080 (2023). <https://doi.org/10.1109/ICOEI56765.2023.10125640>

14. Emuoyibofarhe, J., Ajisafe, D.: Early skin cancer detection using deep convolutional neural networks on mobile smartphone. *Int. J. Inf. Eng. Electron. Bus.* **12**, 21–27 (2020). <https://doi.org/10.5815/ijieeb.2020.02.04>
15. Francese, R., Frasca, M., Risi, M., Tortora, G.: A mobile augmented reality application for supporting real-time skin lesion analysis based on deep learning. *J. Real-Time Image Proc.* **18**(4), 1247–1259 (2021). <https://doi.org/10.1007/s11554-021-01109-8>
16. Gajera, H.K., Nayak, D.R., Zaveri, M.A.: A comprehensive analysis of dermoscopy images for melanoma detection via deep CNN features. *Biomed. Sig. Process. Control* **79**, 104186 (2023). <https://doi.org/10.1016/j.bspc.2022.104186>
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2016). <https://doi.org/10.1109/CVPR.2016.90>
18. Huang, G., Liu, Z., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2261–2269 (2017)
19. Jaikishore, C., Udutalapally, V., Das, D.: AI driven edge device for screening skin lesion and its severity in peripheral communities. In: 2021 IEEE 18th India Council International Conference (INDICON), pp. 1–6 (2021). <https://doi.org/10.1109/INDICON52576.2021.9691666>
20. Kousis, I., Perikos, I., Hatzilygeroudis, I., Virvou, M.: Deep learning methods for accurate skin cancer recognition and mobile application. *Electronics* **11**(9) (2022). <https://doi.org/10.3390/electronics11091294>
21. Polat, Ö., Kartal, M.S.: Detection of benign and malignant skin cancer from dermoscopic images using modified deep residual learning model. *Artif. Intell. Theor. Appl.* **2**, 10–18 (2022)
22. Pereira, P.M., et al.: Melanoma classification using light-fields with morlet scattering transform and CNN: surface depth as a valuable tool to increase detection rate. *Med. Image Anal.* **75**, 102254 (2022). <https://doi.org/10.1016/j.media.2021.102254>
23. Shi, Y., Yang, K., Yang, Z., Zhou, Y.: Model compression for on-device inference, chap. 5. In: Shi, Y., Yang, K., Yang, Z., Zhou, Y. (eds.) *Mobile Edge Artificial Intelligence*, pp. 71–82. Academic Press (2022)
24. Tan, M., Le, Q.V.: EfficientNetV2: smaller models and faster training. In: Meila, M., Zhang, T. (eds.) *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18–24 July 2021, Virtual Event*, vol. 139, pp. 10096–10106. *Proceedings of Machine Learning Research (PMLR)* (2021)
25. Zia Ur Rehman, M., Ahmed, F., Alsuhibany, S.A., Jamal, S.S., Zulfıqar Ali, M., Ahmad, J.: Classification of skin cancer lesions using explainable deep learning. *Sensors* **22**(18) (2022). <https://doi.org/10.3390/s22186915>