ACORDO DE PARCERIA Nº 05/23 FADE/UFPE/SOFTEX - RESIDENCIAL IC13 (CONVÊNIO Nº 02/2023 UFPE) 23076.125530/2022-28

Lux.AI

Image segmentation and detection on clinical dermatology from RGB images

1. INTRODUCTION

Building on research into image-based artificial intelligence applied to clinical dermatology, it is essential to validate the input images before using them for diagnostic support. Through image segmentation or detection, we can isolate the area of interest corresponding to lesions in the images [1]. This approach filters out irrelevant information, focusing on critical regions that capture key lesion attributes.

This report evaluates the leading methods for image segmentation and detection, analyzing their application in dermatology by processing datasets, annotations, and models. We focus on lightweight models [2,3,4,5] that are suitable for mobile applications, aligning with the requirements of our clinical use case.

2. METHODOLOGY

Image Segmentation

To evaluate image segmentation in clinical dermatology, we employed the **UNet** [6] architecture, which is one of the most widely used frameworks for medical imaging tasks. Following the original design, we implemented the architecture in PyTorch, as illustrated in **Figure 1**.

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

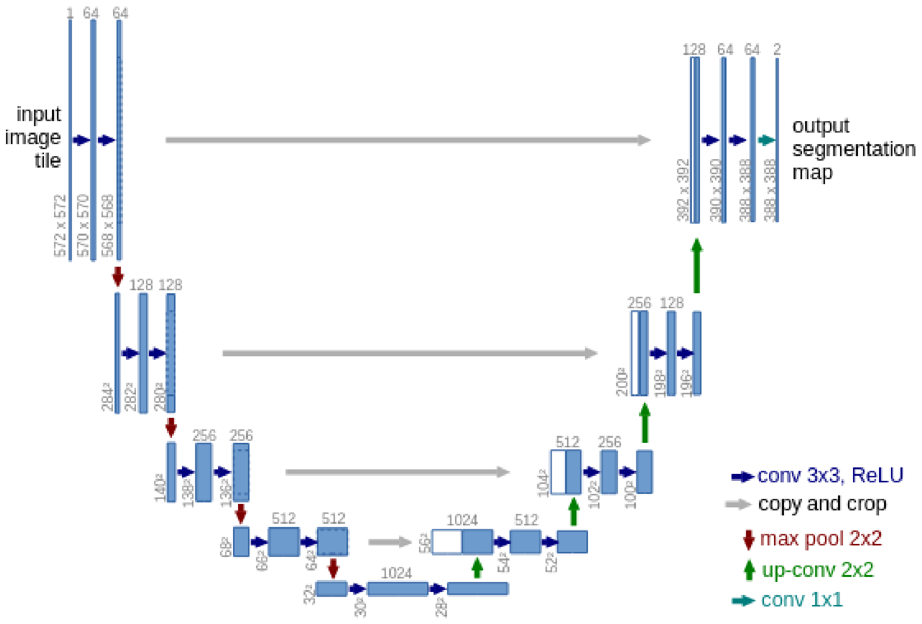


Figure 01 - UNet architecture. Each blue box corresponds to a multi-channel feature map. White boxes represent copied feature maps. The arrows denote the different operations. Image obtained from [6]

In the scope of this project, we aim to utilize the evaluated models in end-user applications to assist patients and generalist doctors prior to consultations with specialized dermatologists.

Therefore, we also assessed the **UNeXt** model [2], illustrated in **Figure 2**, which is designed to operate on simpler devices, such as smartphones. The UNeXt architecture utilizes a tokenized MLP structure, enabling the shifting of previous activations and reducing the feature maps passed through the network. By incorporating tokens, the model becomes simpler, yet it demonstrates the ability to maintain accuracy, even surpassing the original UNet model. We then compared the results obtained from both the UNet and UNeXt models, focusing on accuracy and execution performance.

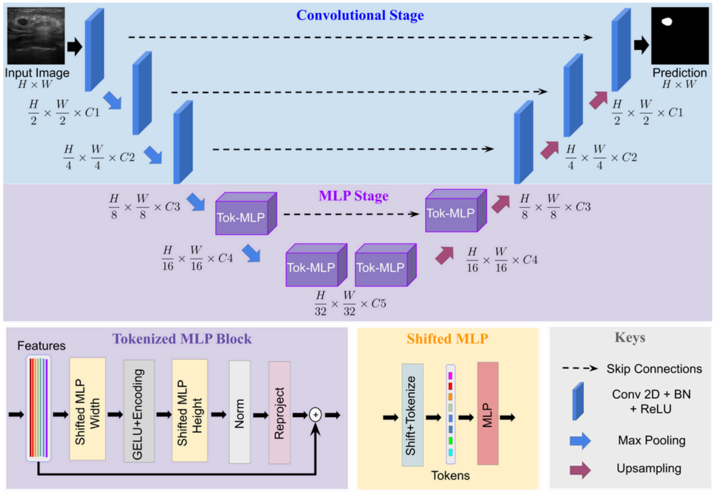


Figure 02 - UNeXt architecture. Image obtained from [2]

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

The primary task is to perform image segmentation; however, few datasets provide this information due to the extensive annotation effort required and the need for validation by specialists. **Figure 3** shows an example of image segmentation ground truth from the **ISIC18** dataset [11]. Fortunately, the ISIC challenges up until 2018 included segmentation masks alongside their respective color images. Thus, we utilized the most recent version of the dataset (**ISIC18**) to train and validate our models. As this dataset is widely referenced in the literature, including in the original UNet and UNeXt papers, it allows us to test and compare our implementation with the original reported models.

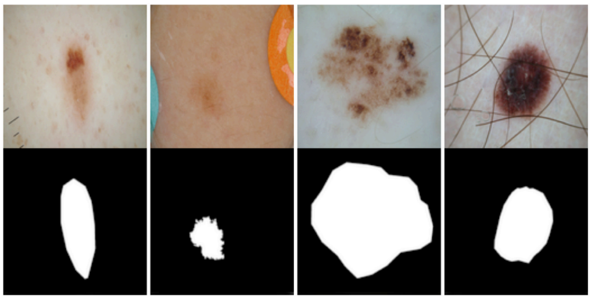


Figure 3 - Example of ISIC18 samples for the segmentation task. Image obtained from [11].

Object Detection

Due to the limited data available for lesion segmentation, we also evaluated various object detection architectures to detect skin lesions in images. Detection only requires an approximate target area, allowing us to leverage annotations from different datasets with less specialized validation. Additionally, many object detection architectures are optimized for execution performance and can run on simple mobile devices, aligning with our objective. Also, we extend segmentation labels by taking the maximum and minimum mask's coordinates in the image to draw the lesions bounding box used in detection training.

We tested the recent **YOLOv7** detection architecture [3], a leading state-of-the-art approach, and compared it to detection models based on **MobileNetV2**, **MobileNetV3**, [4,5] and **EfficientDet** variants [7].

Since detection labels are more commonly available than segmentation masks, we could utilize multiple datasets provided through the **RoboFlow API**[1], which are licensed for research use. By implementing these detection strategies, we can use the segmentation or detection model as a filter to accept or reject images based on the presence of skin lesions, enabling more focused classification in subsequent steps.

Given that we are working with dermatology images in clinical settings, we incorporated variations by validating the use of augmentations to enhance model generalizability across diverse challenges, including capture blur, lighting conditions, and image noise. Finally, we converted the trained models to **ONNX**[2] format, allowing them to be deployed in our mobile application using **TFLite**[3] for efficient inference.

[1] roboflow.com
[2] onnx.ai/
[3] www.tensorflow.org/lite

Metrics

To evaluate model performance in image segmentation, we use the Intersection over Union (IoU) score [8], a widely applied metric in detection and segmentation tasks. IoU measures the overlap ratio between ground truth labels and predicted values, quantifying the intersection between the ground truth mask and the predicted mask. Additionally, we report the Dice score [9] for segmentation, which is conceptually similar to IoU but incorporates the size of the sets, normalizing the score to account for the relative sizes of elements. For both IoU and Dice scores, higher values indicate better performance.

For detection tasks, we evaluate performance using mean Average Precision (mAP) [10], calculated over the IoU score. Since detection models generate multiple bounding boxes, mAP computes the mean precision values for predictions that closely align with the ground truth. While segmentation and detection scores (e.g., pure IoU vs. mAP) cannot be directly compared, mAP provides a reasonable approximation for assessing the effectiveness of detection techniques relative to segmentation methods.

3. EXPERIMENTS AND DISCUSSIONS

Image Segmentation - UNet and UNeXt comparison

Table 1 - Image segmentation results on ISIC18 for UNet and UNeXt.

Model	Reported Accuracy (IoU)	Obtained Accuracy (IoU)	Obtained Accuracy (DICE)	Inference time	Weight size (MB)
UNet	0.7455	0.7168	0.7163	13.21s (CPU) 08.80s (GPU)	1.40MB
UNeXt	0.8170	0.7870	0.8716	110ms (CPU) 5.6ms (GPU)	5.9MB

As shown in table 1, the achieved accuracy is comparable to the results reported in the reference works. UNeXt outperforms UNet in terms of accuracy and offers faster runtime, despite having larger weights. Both networks demonstrate accuracy close to state-of-the-art, but UNet requires longer inference times on a CPU, making it less suitable for mobile applications. Besides the use of MLP, which increases UNeXt's storage size, it enhances performance of operations and improves network inference efficiency. While segmenting lesion areas can be highly valuable, only a few datasets provide segmentation masks suitable for training. Within the scope of ISIC18 (dermoscopic images), model inference performs adequately. However, when tested qualitatively on a clinical dataset aligned with our final application, the results were less satisfactory. As shown in Figure 4 and 5, the predicted masks do not perform as well as those on dermoscopic data like ISIC18.To address this limitation, we can extend the model's knowledge through fine-tuning on

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

clinical datasets. However, this requires the availability of clinical segmentation masks during training, which is essential for improving performance in real-world clinical scenarios.

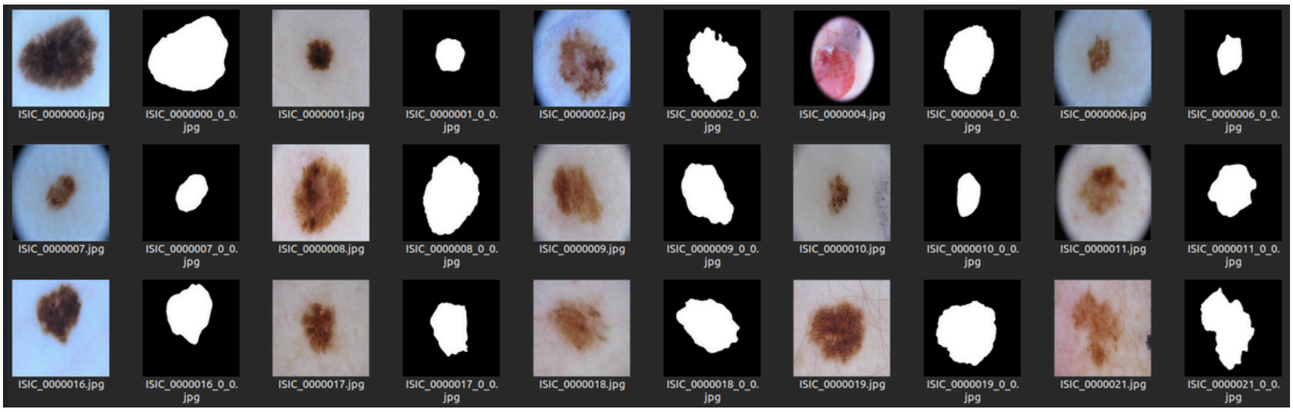


Figure 4 - Segmentation results on ISIC18 dermoscopic dataset.

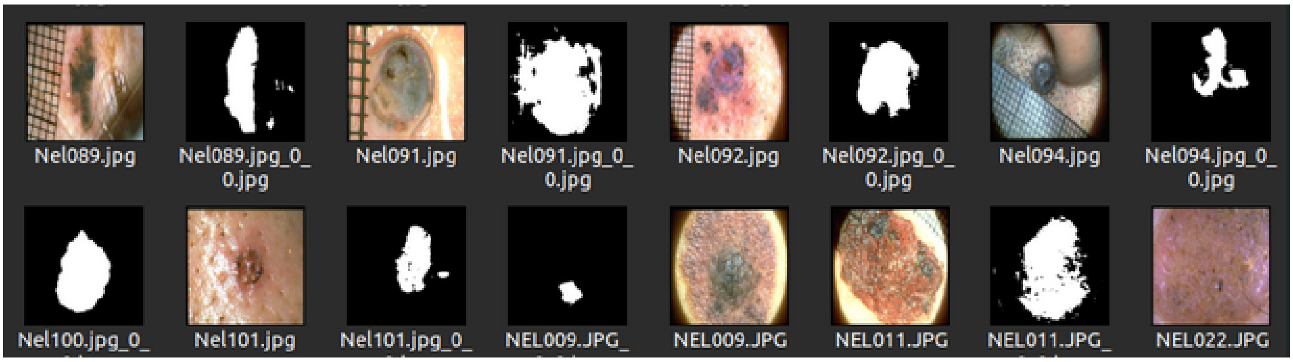


Figure 5 - Segmentation results on derm7pt clinical dataset.

Datasets merge and conversion for object detection

As mentioned before, segmentation dataset is limited and acquiring mask labels is a cumbersome task. Thus, to improve model training on the lesions localization, we collected publicly available datasets containing bounding box annotations for image detection. These annotations enabled us to train lesion detection models using images from multiple sources. The following datasets were downloaded and merged:

1. Skin Cancer Computer Vision Project (https://universe.roboflow.com/skin-cancer-yp3qt/skin-cancer-svnul), 5 classes, 5220 images, license CC BY 4.0
2. Melanoma cancer Computer Vision Project (https://universe.roboflow.com/universitas-islam-nahdlatul-ulama/melanoma-cancer), 3 classes, 225 images, license CC BY 4.0
3. Kuchbhe Computer Vision Project (https://universe.roboflow.com/hasnain-uhyxo/kuchbhe), 7 classes, 1486 images, license ODbL v1.0
4. Zq Computer Vision Project (https://universe.roboflow.com/shizhen/zq-uqc77), 7 classes, 3983 images, license CC BY 4.0

[1] roboflow.com
[2] onnx.ai/
[3] www.tensorflow.org/lite

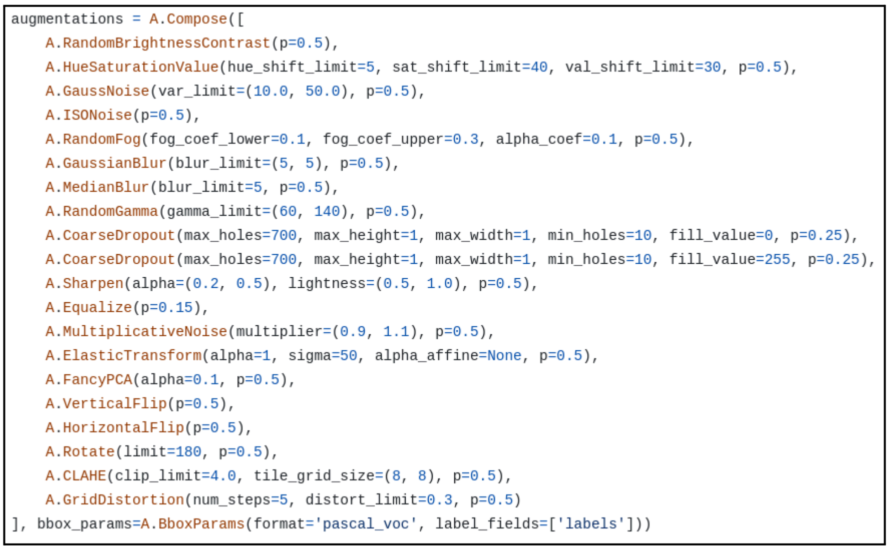
5. Melanoma Detection Computer VIsion Project (https://universe.roboflow.com/wayamba/melanomadetection-l1n4s), 2 classes, 2051 images, license CC BY 4.0

By consolidating these datasets, we aimed to create a robust training set, ensuring diversity in image sources, lesion types, and demographic characteristics. This allows for the development of models capable of detecting lesions across different contexts while increasing the number of samples to enhance training. Since a single dataset is often insufficient to effectively adjust model parameters, combining multiple datasets ensures better generalization and robustness.

To focus on detecting lesion boundaries, we simplified the classification task into two classes. This binary classification approach helps in the detection process, concentrating the model's efforts on distinguishing lesion-containing regions from non-lesion areas. It ensures that the model learns to detect boundaries accurately, regardless of lesion type or dataset source.

Applying Augmentations

As described in the report *“Data Pre-processing and Augmentation on Dermoscopy Images for Skin Lesion Classification”* [12], we experimented with various augmentation techniques to enhance generalization in dermatology classification tasks. Building on this approach, we applied these techniques to lesion detection, adapting and expanding operations to optimize performance for detection models (Figure 6).



```
augmentations = A.Compose([
    A.RandomBrightnessContrast(p=0.5),
    A.HueSaturationValue(hue_shift_limit=5, sat_shift_limit=40, val_shift_limit=30, p=0.5),
    A.GaussNoise(var_limit=(10.0, 50.0), p=0.5),
    A.ISONoise(p=0.5),
    A.RandomFog(fog_coef_lower=0.1, fog_coef_upper=0.3, alpha_coef=0.1, p=0.5),
    A.GaussianBlur(blur_limit=(5, 5), p=0.5),
    A.MedianBlur(blur_limit=5, p=0.5),
    A.RandomGamma(gamma_limit=(60, 140), p=0.5),
    A.CoarseDropout(max_holes=700, max_height=1, max_width=1, min_holes=10, fill_value=0, p=0.25),
    A.CoarseDropout(max_holes=700, max_height=1, max_width=1, min_holes=10, fill_value=255, p=0.25),
    A.Sharpen(alpha=(0.2, 0.5), lightness=(0.5, 1.0), p=0.5),
    A.Equalize(p=0.15),
    A.MultiplicativeNoise(multiplier=(0.9, 1.1), p=0.5),
    A.ElasticTransform(alpha=1, sigma=50, alpha_affine=None, p=0.5),
    A.FancyPCA(alpha=0.1, p=0.5),
    A.VerticalFlip(p=0.5),
    A.HorizontalFlip(p=0.5),
    A.Rotate(limit=180, p=0.5),
    A.CLAHE(clip_limit=4.0, tile_grid_size=(8, 8), p=0.5),
    A.GridDistortion(num_steps=5, distort_limit=0.3, p=0.5)
], bbox_params=A.BboxParams(format='pascal_voc', label_fields=['labels']))
```

Figure 6 - Augmentation applied on detection training

In the additional processing, we applied cropping at various image scales to avoid bias toward detecting lesions of specific sizes. We also introduced positional shifts in bounding box annotations and resized the input images using bilinear interpolation. For training, 80% of the samples were used, with the remaining 20% evenly split into 10% for validation and 10% for testing.

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

Testing and comparing Detection architectures

As shown in Table 2, we trained and tested various architectures on ISIC18. YOLOv7 achieved the highest mAP scores among the evaluated models. However, other architectures, such as SSD MobileNet, offer advantages in terms of storage size and inference speed. SSD MobileNet, for instance, sacrifices only 3 percentage points in mAP compared to YOLOv7 but reduces storage requirements by more than half. This makes it a strong candidate for scenarios where inference performance and resource efficiency are critical, such as mobile applications. Balancing detection accuracy with hardware constraints, SSD MobileNet is particularly suitable for lightweight, on-device processing.

We emphasize the critical role of data augmentation in this context. When working with small datasets, training solely on the original images often leads to early overfitting, severely limiting the model's generalization capabilities. In our experiments, training without augmentation resulted in detection scores between 0.01 and 0.05 mAP. However, using the same setup with data augmentation significantly improved the results, achieving mAP values in the range of 0.4 to 0.6. This demonstrates that augmentation is essential for enhancing performance and mitigating the challenges posed by limited datasets.

Table 2 - Detection score training different networks.

Model	mAP	Size (Float32)	Size (Float16)
YoloV7	0.601	146MB	73.1MB
SSD MobileNetV1	0.572	26.3MB	13.2MB
SSD MobileNet lite V2	0.532	12.2MB	6.1MB
EfficientDet lite0	0.585	4.4MB	-

Fine-Tuning detectors on different datasets

We evaluated the impact of merging datasets and applying fine-tuning to tailor the model to specific scenarios. The YOLOv7 model was tested on a subset of approximately 200 images collected from Hospital das Clínicas UFPE (HC-UFPE) with bounding boxes annotated by doctors and medicine students. Due to the limited size of the HC-UFPE dataset, it was insufficient for standalone training, resulting in a low mean average precision (mAP) of 0.04. By combining the HC-UFPE dataset with the Skin Cancer Computer Vision Project dataset, the mAP improved to 0.247 for multiclass detection, where each lesion type was detected individually. When simplifying the task to binary detection (positive/negative cases), the combined datasets yielded a mAP of 0.519. Finally, merging all available samples from multiple sources significantly boosted detection performance on the HC-UFPE dataset, achieving a mAP of 0.834. This demonstrates the

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

effectiveness of our data-merging approach. By integrating data from sources, we enhanced model generalization and provided sufficient samples for training a robust lesion detection model.

Model	mAP
1 Dataset multiclass (HC)	0.04
Merging 2 datasets multiclass (HC + Skin Cancer)	0.247
Merging 2 datasets 1 class (positive/negative)	0.519
Merging all data 1 class (positive/negative) (HC + all)	0.834

Converting model and using on mobile

Finally, we deployed our best detection model on a mobile application to identify and mark lesions in smartphone images [13]. Using the trained YOLOv7 architecture, we converted the model with the LiteRT framework, enabling its integration into a REACT/TFLite-based application. Since the model was initially trained in PyTorch, it was crucial to ensure proper channel order during conversion, as PyTorch uses a channel-first format while TensorFlow employs a channel-last representation. We provided three versions of the model, differentiated by operand precision: int8, float16, and float32. The lightweight nature of the model made it feasible to use the half-precision (float16) version, achieving a good balance between performance and efficiency on mobile devices. Table 4 below summarizes the model's overall performance when tested on images from the HC-UFPE. A qualitative visualization of the detection results is illustrated in Figure 6.

Table 4 - Result of detection model running in smartphone

Metric	Score
Number of images	205
Inference time (mean)	1.13s
Correct detections (IoU >.5)	143
Incorrect detections (IoU <= .5)	90
False negatives	15
Accuracy	0.62
Precision	0.90
Recall	0.68
F1-Score	0.78

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

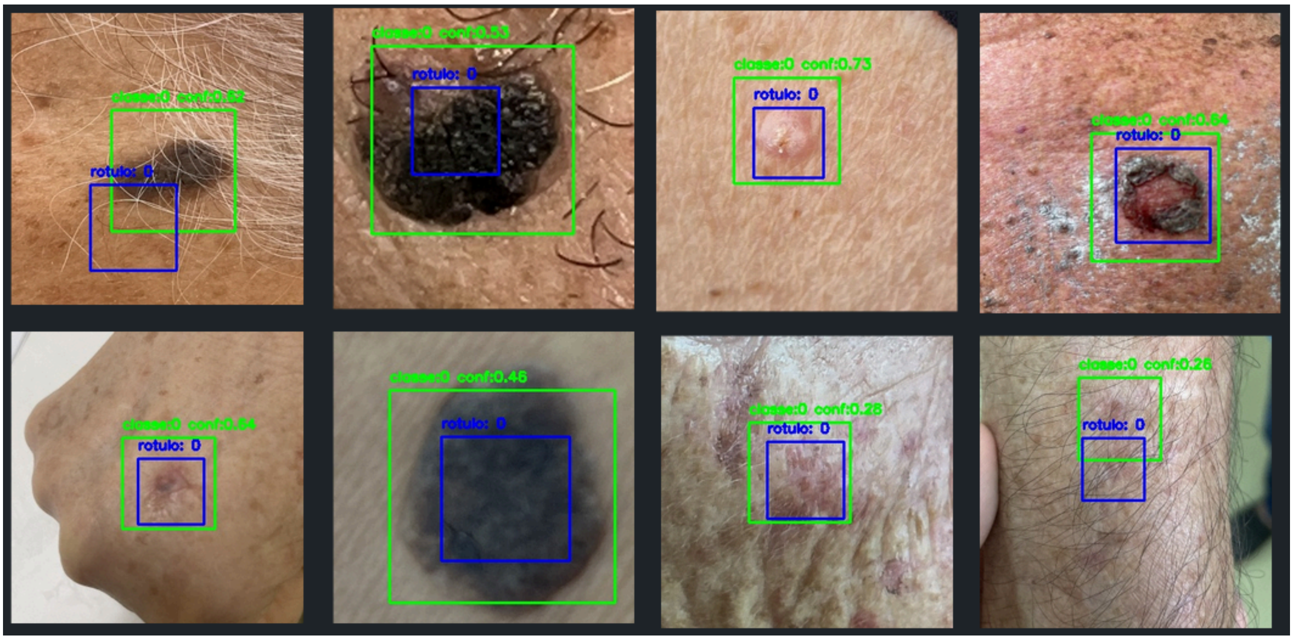


Figure 7 - Visualization of detections running in mobile

4. ACKNOWLEDGEMENTS

This project was supported by the Ministry of Science, Technology and Innovation of Brazil, with resources from Law No. 8,248, dated October 23, 1991, under the scope of the PPI-SOFTEX, coordinated by Softex and published under Residência em TIC 13, DOU 01245.010222/2022-44.

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite

5. REFERENCES

[1] Azad, Reza, et al. "Medical image segmentation review: The success of u-net." *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).

[2] Valanarasu, Jeya Maria Jose, and Vishal M. Patel. "Unext: Mlp-based rapid medical image segmentation network." *International conference on medical image computing and computer-assisted intervention*. Cham: Springer Nature Switzerland, 2022.

[3] Wang, Chien-Yao, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2023.

[4] Howard, Andrew G., et al. "MobileNets: efficient convolutional neural networks for mobile vision applications (2017)." *arXiv preprint arXiv:1704.04861* 126 (2017).

[5] Liu, Wei, et al. "Ssd: Single shot multibox detector." *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*. Springer International Publishing, 2016.

[6] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer International Publishing, 2015.

[7] Tan, Mingxing, Ruoming Pang, and Quoc V. Le. "Efficientdet: Scalable and efficient object detection." *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*. 2020.

[8] Zhou, Dingfu, et al. "Iou loss for 2d/3d object detection." *2019 international conference on 3D vision (3DV)*. IEEE, 2019.

[9] Bertels, Jeroen, et al. "Optimizing the dice score and jaccard index for medical image segmentation: Theory and practice." *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part II 22*. Springer International Publishing, 2019.

[10] Henderson, Paul, and Vittorio Ferrari. "End-to-end training of object class detectors for mean average precision." *Computer Vision–ACCV 2016: 13th Asian Conference on Computer Vision, Taipei, Taiwan, November 20-24, 2016, Revised Selected Papers, Part V 13*. Springer International Publishing, 2017.

[11] Noel Codella, Veronica Rotemberg, Philipp Tschandl, M. Emre Celebi, Stephen Dusza, David Gutman, Brian Helba, Aadi Kalloo, Konstantinos Liopyris, Michael Marchetti, Harald Kittler, Allan Halpern: "Skin Lesion Analysis Toward Melanoma Detection 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC)", 2018; https://arxiv.org/abs/1902.03368

[12] Data Pre-processing and Augmentation on Dermoscopy Images for Skin Lesion Classification. LuxAI. Technical Report, 2024. https://luxai.cin.ufpe.br

[13] Detection in Mobile. LuxAI, 2023.
https://github.com/TIC-13/conversao_e_testes_modelos_de_deteccao_mobile

1. roboflow.com
2. onnx.ai/
3. www.tensorflow.org/lite