Applications of Deep Learning to Measure Skin Lesions

Cecília de Oliveira Penha[1]

[1]Universidade Federal da Fronteira Sul (UFFS)

Abstract. *When treating skin lesions, accurate assessment of wound healing remains a challenge for health professionals. Precise estimation of the wound area is required to determine if the current approach is efficient or if a new intervention is required. This paper proposes a study on the ability of computer vision algorithms and deep learning networks to measure the size of dermatological wounds based on images. By employing an approach based on the U-Net architecture, the study evaluates the model's ability to segment wounds and provide area estimations. Results demonstrate that the proposed method can support decision making by health professionals by offering consistent measurements, contributing to faster and efficient patient care.*

1. Introduction

When treating a skin lesion, regardless of its cause or diagnosis, the assessment of progress through measurement of the affected area is considered an essential step. Measuring the lesion area over time allows health professionals to determine whether the chosen method is efficient for treatment. For instance, if a wound has not healed after a period of four weeks or once aftercare has been provided, it can be classified as a chronic wound, leading to further discomfort and expenses to the patient and the professionals [Martins-Green 2023].

Wound assessment demands time and resources from health professionals, both during the diagnosis and the treatment phase. Over the years, many algorithms have been developed to help health professionals measure wound size or classify a lesion with greater precision. Computer vision based technologies are able to detect wound areas using pixel classification, for example, and classify them using segmentation based approaches, made possible by convolutional neural networks [Reifs et al. 2023].

This paper proposes an approach to wound size measurement using deep learning, specifically the U-Net neural network, which is widely utilized for medical image segmentation and analysis due to its ability to combine accuracy and complex feature extraction through resolution changes[Liu et al. 2021].

The study resulted in the development of an algorithm that utilizes the U-Net for measurement, the YOLO(You Only Look Once) network for object detection, and the conversion of the wound size from pixels to centimeters. The study was conducted using the HAM-10000 dataset. Based on the YOLO network's model for object detection, the program was trained on the medical images provided by the dataset to identify skin lesions, which were then measured with the use of the U-Net network. Finally, a square shaped figure was used as a scale from pixels to centimeters.

The following sections will present a detailed explanation of how the study was conducted. Section 2 will present the theoretical basis for this study. Section 3 will present a few related works that were used as a basis for this study. Section 4 will describe how the

model was trained and how the algorithm was developed. Finally, Section 5 will present the results and metrics obtained. Section 6 will present the conclusion.

2. Theoretical Basis

2.1. Deep Learning

Deep Learning is one of the branches of Machine Learning. It received its name due to the number of layers it requires to work, resembling a deep network of connections. The purpose of this is to simulate a human brain and its connections [Voulodimos et al. 2018], in order to process data and make decisions. The recent developments in technologies such as GPUs (Graphics Processing Unit), along with availability of labeled datasets, have transformed the research in deep learning fields, such as object detection, motion tracking, object classification, and computer vision areas in general.

The focus of this paper is to explore deep learning in the medical field, which has greatly benefited from the volume of data made public in high-quality datasets. Thanks to this fact, models can be trained for tasks such as diagnosis and measuring, becoming a powerful tool for medical professionals. Labeled datasets (that contain both the image and class) are critical for supervised learning, which can be fine tuned for use with transfer learning for specific tasks [Esteva et al. 2021].

2.2. Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are a branch of the deep learning field, frequently applied to computer vision algorithms. This is due to the CNN's ability to detect features across multiple layers through techniques such as resolution changes, which allows it to tackle more complex images [Mienye et al. 2025].

CNNs are powered by layers of filters distributed across a grid-like structure (such as an image), moving from the raw input to complex feature extractions. CNNs have been adopted for tasks such as image classification (diagnosis), in fields such as oncology, neurology, and dermatological lesions, with results often outperforming experts. One major example was the GoogleNet, used to detect cancer with 89% accuracy, as opposed to the 70% achieved by human professionals [DR. and RV. 2022].

2.2.1. U-Net

The U-Net is one of the most commonly utilized CNN structures when it comes to medical imaging [Mienye et al. 2025]. It is named after its U-shaped structure, which employs layers of encoding - extracting high-level features by reducing the image resolution - and decoding, which reconstructs the original image.

One of the greatest strengths of the U-Net is its pixel-level segmentation, allowing for higher precision when it comes to detecting borders [Chang et al. 2024]. Also, U-Net makes use of skip connections, which allows for communication between layers and back propagation [Liu et al. 2021]. Because of this, U-Net is widely used in the medical field, extracting complex features even with small datasets. The purpose of this paper is to provide a study of U-Net's ability to measure skin lesions based on the HAM-10000 dataset, combined with YOLO network for object detection.

2.2.2. YOLO

The YOLO(You only look once) network is one of the most widely used object detection models in literature. Its speed and accuracy allow for it to be used in real-time environments, thanks to its single-shot approach. YOLO divides the image into a grid, making predictions directly within its cells [Vijayakumar and Vairavasundaram 2024].

While YOLO was first released in 2016, the 2023 YOLOv8 was chosen for this study, which includes two convolution operations and one bottleneck. It incorporates data augmentation methods, such as MixUP(combining two datapoints) [Vijayakumar and Vairavasundaram 2024], allowing for better metrics. However, since YOLO is not trained on dermatoscopic images, a custom model was required for this study.

2.3. Wound Measuring

Over the past decade, many studies have explored the possibilities of integration between computer vision and medicine, especially wound segmentation [Reifs et al. 2023]. However, for proper diagnosis and follow-up treatment, it is necessary to measure the wound frequently, in order to evaluate its healing progress. During the wound measuring process, images are often required to assess progress. This has led to a development of publicly available data, making it possible to train models based on dermatological datasets.

Current methods for wound measuring, such as planimetry (tracing the wound), can be time consuming and have high error rates [Reifs et al. 2023]. For this reason, it is important to explore how technology can improve this process, in order to help patients receive faster care. Currently, there are a few mobile approaches to this problem, such as the *imitoMeasure* App [K. et al. 2025], developed based on a rabbit wound healing study, which attempts to reproduce planimetry using an image with a sticker-based scale.

3. Related Works

3.1. Application of deep learning in wound size measurement using fingernail as the reference

The paper [Chang et al. 2024] describes an experiment to measure skin lesions using deep learning, with a fingernail included in the picture for reference. Three networks were chosen for the experiment: The Mask R-CNN for detection and measuring of the fingernail, the YOLOv5 to crop the wound, and the U-Net to calculate the final area. The study obtained 100% of fingernail detection and 97.76% of wound detection, with 0.95 mean pixel accuracy for the U-Net, evaluated on 248 images. It also received 90% satisfaction rate from thirty inexperienced users who tested the experiment.

This paper used a square shaped figure drawn on the image as a reference, as opposed to a fingernail physically present as the picture is taken. However, a similar process was used to combine YOLOv8 with the U-Net architecture. The use of a drawn in figure makes it easier for inexperienced users to use this model, as no additional actions are required for measuring the lesion.

3.2. A Mobile App for Wound Localization Using Deep Learning

The paper [Anisuzzaman et al. 2022] presents an automated wound localizer using the YOLOv3 model, isolating the lesion on the image through a mobile app. The YOLO

model resulted in mean average precision of 93.9%. The study was based around the AZH Wound Database, with 1,010 images split into three classes of ulcers. With the architecture implemented for the mobile app, it is possible to detect the region of interest through live video.

As a contrast, this paper proposes a study for still images, however, with the addition of measurement of the lesion detected, as well as localization. Going beyond detecting the bounding box for the lesion, the study also results in detailed pixel by pixel segmentation of the lesion in the picture, as well as using a more recent version of YOLO architecture.

3.3. Clinical validation of computer vision and artificial intelligence algorithms for wound measurement and tissue classification in wound care

The paper [Reifs et al. 2023] proposes an approach to classifying the region of interest using pixel evaluation techniques, as well as area measurement and tissue classification. Wound detection was done with user collaboration, through drawing the general wound area in order to help the model detect the contours and produce a mask. After applying the mask to isolate the wound, the model divides the pixels in groups according to their similar characteristics, using k-means and clustering. Area is calculated using a blue square shaped sticker for reference.

This paper proposes a similar study, with wound detection and area measurement following a fixed shape as reference. However, the wound detection is done without additional input from the user, and the square used as reference is drawn on the image, as opposed to being part of the picture as it is taken.

4. Methodology

The methodology consisted in four parts: data selection and processing, model implementation, training, and analysis of the results. The data was obtained through the HAM10000 dataset, which contains 10,015 images and their respective masks(Figure 2), divided among 7 skin lesion classes. The model used two types of neural networks: the U-Net for area measuring and the YOLOv8 network for object detection. The script was built using the Python 3.11 language and libraries such as OpenCV and TensorFlow. Hyper parameters chosen were batch size of 32, 30 epochs, and Binary Cross-Entropy + Dice Loss for the loss function, as well as 128 x 128 input, ReLU + sigmoid activation functions, and filters of 64, 128, and 256. The threshold for segmentation was 0.5 in both U-Net and YOLO.

The HAM10000(Human against machine with 10000 training images) was chosen due to its availability and the large number of works it is referenced by. It contains 10015 dermatoscopic images from different populations, as seen on Figure 1, made available by the ISIC archive [Tschandl et al. 2018]. Division was done using the train-test-split method, with 20% for testing and 80% for training. The training portion had its own division, with 70% for training and 30% for validation. The internal division between training and validation allows for evaluation of the model's behavior during the epochs, monitoring overfitting and adjusting of weights.

Normalization of the dataset was done by converting the values from RGB(0-255) to floating point values between 0 and 1, in order to make the data compatible with the

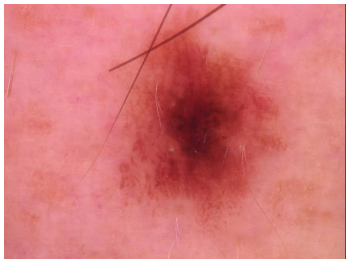

Figure 1. Example of image from dataset

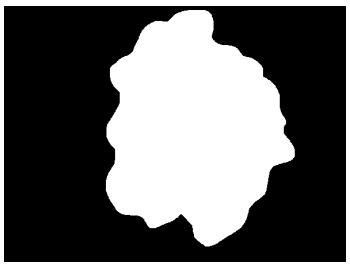

Figure 2. Example of mask from dataset, representing a lesion

sigmoid activation function of the U-Net, as well as the loss functions. Images were resized to 128 x 128 pixels for standardization, using the OpenCV resize function. This was done to reduce computational cost during training, given the large size of the dataset. Masks were resized using the nearest-neighbor method, in order to avoid intermediary values, preserving the binary nature.

The first step was to draw a square shaped reference marker in the upper left corner of the image, as seen in Figure 3, in order to have a fixed scale to measure the lesion area. The square had a known area of one centimeter. By calculating its size in pixels, it was possible to determine a pixel to centimeter conversion factor, allowing the lesion area to be expressed in real-world units. The figure was not drawn on the mask, therefore, the U-Net would not see it as part of the lesion and it would not generate interference in training. The square was also not included in the original dataset. Instead, a copy of the image was used.

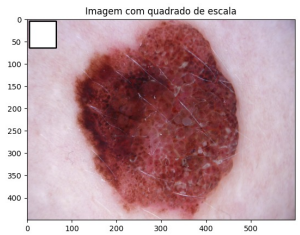

Figure 3. Example of image with square

The next step was the implementation of the U-Net model. For this experiment, a 128 x 128 input size was chosen, in order to balance computational cost and the level of detail, along with two encoder blocks, one bottleneck, and two decoder blocks. The layers went from 64, to 128, to 256 filters.

The encoder section consisted of two 3 x 3 convolution blocks, followed by the ReLU function, which suppresses negative inputs, and Max Pooling to reduce the feature map's resolution. The decrease in resolution combined with the increase in filters allowed for deeper feature extraction. The bottleneck section applied the additional 256 filters, allowing for further feature extraction before reconstruction. Finally, the decoder block reconstructed the map by upsampling the resolution. Skip connections were incorporated between each decoder block and its respective encoder activation, allowing the model to recover information that could otherwise be lost during downsampling. Figure 4 presents the raw output of the U-Net.

The output was a 1 x 1 convolution activated by the sigmoid function, which resulted in a binary segmentation mask representing the lesion. The mask was built based on a probability map with values between 0 and 1, each representing the likelihood of the respective pixel being part of the lesion. The final mask was obtained by thresholding those values. Figure 5 presents an example of predicted mask.

The loss function chosen was a fusion between Binary Cross-Entropy (BCE) and Dice Loss. BCE has its focus on pixel-wise classification accuracy, penalizing incorrect predictions at individual pixel level. Dice Loss evaluates the overlap between the predicted and the ground truth masks, making it useful to handle class imbalance, such as images with a very small lesion area. Both metrics being integrated allows for more precise pixel classification and shape optimization.

Training was carried out over 30 epochs, with a batch size of 32 and the Adam optimizer. The model examined the training set 30 times, with turns of 32 images. After making a prediction for each image, it calculated loss and adjusted the weights using backpropagation, before moving on to the next batch. The Adam(Adaptive Moment Estimation) optimizer was responsible for understanding the learning rate and adjusting the weights.

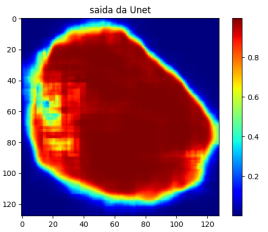

Figure 4. Raw output of the U-Net

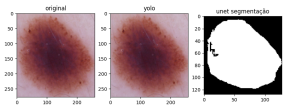

Figure 5. Original image and predicted mask

The following step was incorporating the use of the YOLO network for object detection. The model was first trained on the HAM10000 dataset(Figure 6), and then

integrated with the previously trained U-Net. The U-Net model was applied to every positive detection of the YOLO model(Figure 7). YOLO is a single stage object detection architecture [Wicaksana et al. 2025], designed for high processing speed. Figure 8 presents the final mask after the crop is sent to the U-Net.

Finally, after integrating the networks, the final stage of the pipeline involved detecting the one centimeter reference square included in the pre-processing step. By calculating its area in pixels, the pixel to centimeter scale was determined.

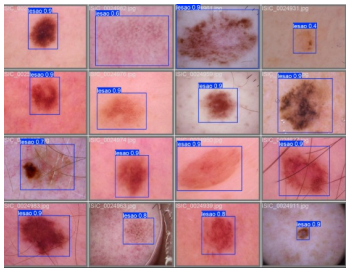

Figure 6. Training of YOLO

Figure 7. Lesion detected by YOLO, square in the corner

Detection of the reference square was done through identification of its color and geometry. The first step was to convert the test image to grayscale, as the process was based around pixel intensity. The threshold function of the OpenCV library was used to isolate the white square in the gray scaled image, detecting pixels with intensity above 230 (in a scale from 0 to 255) as white, while the remaining pixels became black. This produced a mask for the square figure, similar to the lesion mask. Morphology operations were applied to reduce noise, such as filling imperfections or highlighting edges around the square.

After the thresholding and morphological refinement, it was possible to search for the contours of the square along the binary mask. Contours were extracted using the OpenCV *findContours* function and sorted by area, under the assumption that the square would have the largest white region in the image. With this, the program calculated the perimeter and bounding rectangle of the figure.

The program evaluated the geometrical characteristics of the contour. If the shape had four vertices detected and similar width and height, it was classified as a square. The scale was then obtained by dividing 1 cm² by the number of pixels composing the square.

The addition of YOLO for object detection was a choice based around the need for consistency among all sorts of conditions, such as poor lighting or shadows around

the image. After detection, the bounding box coordinates were used to crop the region of interest, leaving only the lesion. The ROI was then normalized to 128 x 128 pixels and [0,1] interval to ensure compatibility with the U-Net model.

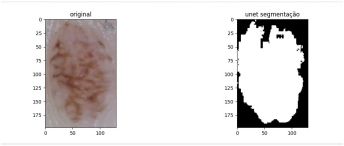

Figure 8. Lesion and predicted mask

The U-Net received the cropped output of YOLO as input and produced a pixel-by-pixel binary mask of the lesion, with black(0) representing the background and white(1) being the wound area. The total area was then calculated by counting the amount of white pixels, followed by conversion to cm² with the use of the previously determined scale. For this process, the image was converted back to its original dimensions, in order to keep consistency of the area. Therefore, the process integrated detection, segmentation and measuring. Finally, visualization was done by exhibiting the U-Net output, as well as the YOLO detection and the predicted mask, as seen in Figure 10.

The U-Net returned values between 0 and 1 for each pixel. The threshold chosen was 0.5, meaning every pixel above the threshold was considered white(1), and the ones below it were considered black(0). Morphology operations were applied to reduce noise - Open to reduce isolated white pixels, and Close to clear imperfections inside the lesion. The mask was then resized back to its original dimension, instead of 128 x 128, in order to calculate the actual area. This was done using the OpenCV inter-nearest method, to keep values between 0 and 1. Then, the model calculated the sum of all pixels, therefore all those that were white, or with a value of 1. This returned all pixels marked as belonging to the lesion. The final step was to use the square reference to convert the pixel area to cm², as shown by Figure 9.

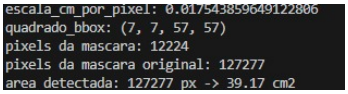

Figure 9. Example of area result

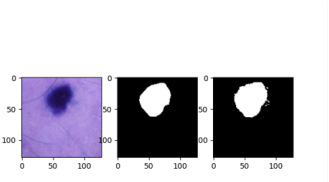

Figure 10. Original image, real mask and predicted mask

5. Results

The following section will cover the results obtained after the training of 30 epochs of the model, analyzing accuracy, precision and recall, as well as loss, obtained through

the BCE+Dice Loss function. Metrics were obtained through TensorFlow. Accuracy represents the total percentage of correct pixel predictions, usually high due to the large amount of black pixels in the mask, corresponding to the background or non-lesion area of the image. Precision represents the percentage of pixels correctly marked as part of the lesion by the predicted mask, while recall indicates how many of the pixels that should be positive have been detected. Keeping training and validation inputs separate ensures more accurate evaluation of metrics.

Table 1. Metrics of the U-Net Model

Metric	Value
Loss	0.1494
Accuracy	0.9402
Precision	0.8958
Recall	0.8952

The YOLO model was evaluated using mAP50(mean average precision over 50% threshold). For this metric, the predicted bounding box is considered correct if its overlap with the real box reaches at least 50% of accuracy. The model obtained a result of 0.976, indicating that it was able to learn spatial patterns across the dataset with excellent efficiency, as well as localize lesions with high reliability. This performance reflects not only accurate bounding box placement but also consistent detection across the different classes of lesions present in the dataset. Figure 11 shows the graphic representation of metrics.

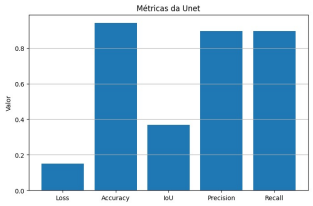

Figure 11. Metrics representation

5.1. Discussion

The results obtained by the U-Net indicate that it was able to maintain consistent learning throughout training, learning the relationship between masks and corresponding lesions accurately, as shown by the low loss margin (0.149). Global accuracy of 0.94 showed a high number of correctly classified pixels, however, given the strong class imbalance of medical images, where background pixels greatly outnumber the lesion, this metric is insufficient for evaluation.

However, precision and recall demonstrate that the model did present adequate results identifying the lesion and avoiding false positives. High precision indicates most pixels marked as positive, or part of the lesion, were correct, while high recall shows that there was little loss in relevant areas.

The loss function is the metric used to quantify the error margin of the model during training. In this experiment, it compared the real mask(ground truth) to the mask

predicted by the U-Net. Loss value across the epochs became smaller at a stable rate, indicating the model avoided over and underfitting [Salman and Liu 2019].

Because the U-Net model was implemented with the YOLO crop as input, its performance was directly influenced by the quality and precision of those bounding boxes. The ability of the U-Net to consistently produce masks in a coherent form demonstrates that the object detection model obtained reliable results, as reflected by its mAP score. Any inaccuracies during the detection stage, such as improperly placed boxes, could affect the segmentation stage and the U-Net's metrics. Therefore, maintaining clean and accurate crops was essential for perfomance.

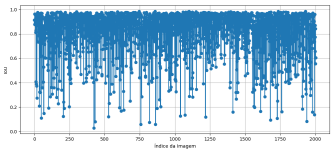

Figure 12. Image by image IoU representation

Most IoU - intersection over union - scores of the U-Net were shown to be in the interval of [0.8, 0.95], which confirms accurate segmentation and construction of predicted masks, as well as no detection of the reference square as part of the lesion. However, lower scores can be explained by improper masks contained in the dataset, as small imperfections can cause IoU to drop considerably, as well as patterns that may confuse the model, such as hairs, shadows, or reflections.

The average IoU of the model, calculated based on all images, reached a value of 0.86. However, global IoU resulted only in a score of 0.36, as it was calculated using internal keras methods with the sum of all True Positives, False Positives and False Negatives. This can result in poor results, as small mistakes stack up over time. Figure 12 shows image by image IoU average.

5.2. Comparison to Related Works

When evaluated in comparison to the works cited before, the model obtained significant results. The 0.976 mAP result of YOLO is higher than the 0.822 obtained by [Chang et al. 2024] with the finger nail, as well as the 0.939 of [Anisuzzaman et al. 2022] with the live localization. As for IoU, the 0.36 score of the model was below the 0.5 score of both works. The precision score of 0.89 was below the 0.9 of [Anisuzzaman et al. 2022].

5.3. Limitations

Low IoU (intersection over union) of 0.36 shows some limitations of the model. IoU measures the predicted mask over the real mask. It is stricter as a metric, because it measures how much of the shape and position of the lesion are correct. Single pixels out of place will drop the final IoU metric, therefore a lower result compared to precision and recall is expected.

Despite the relatively low value of IoU, it is observed that the model achieved what it was proposed - highlighting the area of interest and identifying the mask of the lesion.

Due to the low resolution during the training section, mistakes around the borders of the lesion can be expected. The model's efficiency was also affected by the limited diversity of the dataset in terms of lighting, size or texture. Therefore, increasing the dataset may also prove beneficial for results.

6. Conclusion

In conclusion, the combined approach using YOLO for localization and the U-Net for segmentation proved effective for identifying and measuring skin lesions. The high mAP score obtained by YOLO demonstrated strong detection ability, while the U-Net achieved consistent pixel-level classification, as evidenced by precision and recall scores, despite class imbalance. The integration of the square based scaling method enabled estimation of area in real-world units, indicating the possibility of implementation of automated measurement as a tool for health professionals.

6.1. Further Work

For future works, improvements could be made to enhance the model's performance, such as expanding the training dataset with even more diverse lesion classes and imaging conditions, as well as employing more advanced segmentation architectures, such as U-Net++ [Zhou et al. 2018]. Additionally, using a single-stage pipeline could further improve the model's scores.

Another possible direction is to improve the square detection algorithm, perhaps through machine learning, as opposed to the thresholding based process. Finally, deploying the model as a real environment, especially mobile applications, could bring further results, such as usability studies with professionals.

References

Anisuzzaman, D. M., Patel, Y., Niezgoda, J. A., Gopalakrishnan, S., and Yu, Z. (2022). A mobile app for wound localization using deep learning. *IEEE Access*.

Chang, D., Nguyen, D., and Nguyen, T. (2024). Application of deep learning in wound size measurement using fingernail as the reference. *BMC Med Inform Decis Mak*.

DR., S. and RV., K. (2022). Convolutional neural networks in medical image understanding: a survey. *Evol Intell*.

Esteva, A., Chou, K., and Yeung, S. (2021). Deep learning-enabled medical computer vision. *npj Digital Medicine*.

K., S., SA., B., and M., M. (2025). Comparative assessment of smartphone-based digital planimetry for wound area measurement. *Narra J*.

Liu, X., Song, L., Liu, S., and Zhang, Y. (2021). A review of deep-learning-based medical image segmentation methods. *Sustainability*, 13(3).

Martins-Green, M. (2023). Cutaneous chronic wounds: A worldwide silent epidemic. *Open Access Government*.

Mienye, I. D., Swart, T. G., Obaido, G., Jordan, M., and Ilono, P. (2025). Deep convolutional neural networks in medical image analysis: A review. *Information*, 16(3).

Reifs, D., Casanova-Lozano, L., Reig-Bolaño, R., and Grau-Carrion, S. (2023). Clinical validation of computer vision and artificial intelligence algorithms for wound measurement and tissue classification in wound care. *Informatics in Medicine Unlocked*, 37:101185.

Salman, S. and Liu, X. (2019). Overfitting mechanism and avoidance in deep neural networks.

Tschandl, P., Rosendahl, C., and Kittler, H. (2018). The ham10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions. *Sci Data*.

Vijayakumar, A. and Vairavasundaram, S. (2024). Yolo-based object detection models: A review and its applications. *Multimed Tools Appl*.

Voulodimos, A., Doulamis, N., Doulamis, A., and Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018(1):7068349.

Wicaksana, I., Pramunendar, R., Saraswati, G., and Trisnapradika, G. (2025). Skin lesion classification using yolov11 on the ham10000 dataset. *Jurnal Ilmiah Bidang Teknologi Informasi dan Komunikasi*.

Zhou, Z., Siddiquee, M. M. R., Tajbakhsh, N., and Liang, J. (2018). Unet++: A nested u-net architecture for medical image segmentation.