Assessing the Trade-off Between Model Behavior and Performance Metrics in Wound Image Segmentation

A. Louvart[1], B. Guivarch[1], V.-D. Nguyen[2], W. Haertle[2], K. Bensafia-Cherfa[2], C. Hoffmann[3], A. Al-Jumaily[4], A. Mansour[1]
1. LabSTICC, UMR 8265, ENSTA IP Paris, Brest, France.
2. UVASC LAB, Pont l'Abbé, France.
3. CTB HNIA Percy, Clamart, France.
4. Melbourne Institute of Technology, Sydney, Australia.
Email: arthur.louvart@ensta.fr

***Abstract*—Deep learning has shown significant potential in automating wound segmentation from medical images. However, several models have been optimized in order to align with basic validation metrics, such as the average Dice score or the Intersection over Union (IoU), often with a focus on pixel-level accuracy at the expense of generalization. This focus may result in overlooking a model's true ability to adapt to diverse and complex cases. Moreover, since segmentation masks are inherently ambiguous, a model might achieve a high Dice score while failing to learn meaningful patterns—resulting in less coherent segmentations despite slightly higher average accuracy. In this study, we develop a segmentation model based on EfficientNetV2 with a standard U-Net-inspired decoder and train it on a carefully curated dataset designed to provide high-quality examples with clear segmentation patterns. Our approach achieves state-of-the-art performance, with a Dice score reaching up to 94%. Using k-fold cross-validation on 1,356 clinically validated segmentations, we compare models selected by the best validation score at 100 epochs and the best training score at 30 and 100 epochs after merging validation with training data. Our findings indicate that deploying a validation dataset might not bring the expected optimization. In contrast, manually limiting the number of training epochs may play a crucial role in preserving meaningful segmentation performance.**

***Index Terms*—deep learning, medical image segmentation, evaluation metrics, generalization, edge cases analysis, cross-validation**

I. Introduction

Wounds are disruptions to the body's tissues caused by trauma, surgery, or medical conditions. They are classified as acute or chronic, each with distinct healing characteristics [1]. Acute wounds—such as cuts, abrasions, lacerations, and puncture wounds—typically heal within weeks without major complications [2]. In contrast, chronic wounds, including diabetic, vascular, or pressure ulcers, persist beyond the expected healing period [3], often due to factors like poor circulation, diabetes, aging, or certain medications. These chronic wounds significantly impact patients' quality of life, leading to pain, mobility limitations, and increased healthcare costs [4].

Accurate wound segmentation is crucial for assessing wound size, monitoring healing, and guiding treatment. However, manual segmentation is time-consuming and prone to inter-observer variability, making it inefficient in clinical practice. This has led to growing interest in automating the process to improve efficiency and consistency [5].

Early studies demonstrated the feasibility of using deep learning, particularly convolutional neural networks (CNNs), for wound segmentation. The authors of [6] applied CNNs to diabetic foot ulcer (DFU) segmentation, achieving Dice scores as high as 90%. Building on these promising results, CNN-based architectures have since gained widespread adoption in medical image analysis, particularly for wound segmentation. Models like U-Net, designed for semantic segmentation tasks, have achieved high accuracy in delineating wound boundaries. In [5], the authors reported Dice scores exceeding 90% in general wound segmentation, including DFU, pressure ulcers (PU), and venous leg ulcers (VLU). In [7], they achieve a Dice score of 92% using five-fold cross-validation on 2,372 images of various wound types. While these results demonstrate high performance, deep learning models often lack transparency in the underlying learned concepts, which may lead to an overemphasis on simplistic metrics [8].

Although high Dice scores are commonly viewed as indicators of good generalization, optimizing for these metrics alone can lead to unintended model behavior. The model may latch onto spurious patterns that mainly fit the training data, resulting in unexplainable gaps or incomplete shapes in the resulting segmentation. Moreover, manual segmentation is inherently uncertain and contains variability that the model should not blindly replicate. As such, while optimizing for the highest Dice score can help guide the model towards a solid foundation, pushing for marginal improvements in the score beyond a certain threshold may undermine the model's reliability and generalization.

This study aims to develop a model that achieves consistent generalization while minimizing the occurrence of bad segmentation cases. To this end, we design a model based on EfficientNetV2 [9], pre-trained on the ImageNet-21k dataset[1] [10]. Before fine-tuning, the dataset is carefully curated and normalized (see Section II) to align with target concepts that

[1]ImageNet-21k is a large-scale dataset containing approximately 14 million images across 21,000 classes, serving as an extended version of the widely used ImageNet-1k dataset.

the model should focus on, and avoid the introduction of poor-quality examples. We analyze the progression of poorly segmented images using Dice score histograms and assess segmentation performance through k-fold cross-validation [11]. The segmentations are reviewed individually, providing insights into acceptable segmentation boundaries and offering a deeper understanding of the model's true learning capabilities.

II. Dataset

In this section, we describe the dataset used for training and evaluating our model.

The images and masks used in this study were sourced from a publicly available dataset on Kaggle [12], curated by a platform user. The dataset comprises 2,760 images collected from various public sources, with a portion manually annotated by the creator. While it lacks clinical validation and detailed information on the imaging device, it provides a valuable resource to train deep learning models. 374 images come from Medetic [13], 1,210 from the Foot Ulcer Segmentation Challenge (FUSC) [6], and 1,176 from WSNet [14].

The dataset includes a variety of wounds and related injuries, such as dehisced abdominal wounds, thermal injuries, foot wounds and ulcers, venous and arterial leg ulcers, malignant wounds, meningococcal meningitis wounds, orthopedic wounds, pressure ulcers, and pilonidal sinus wounds. Many of these injuries contain fibrin, granulation, and callus tissue. However, callus is particularly challenging for segmentation tasks due to its visual similarity to skin, as shown in Fig. 1. Consequently, it is excluded from the retained segmentation masks, to prevent significant general performance degradation.

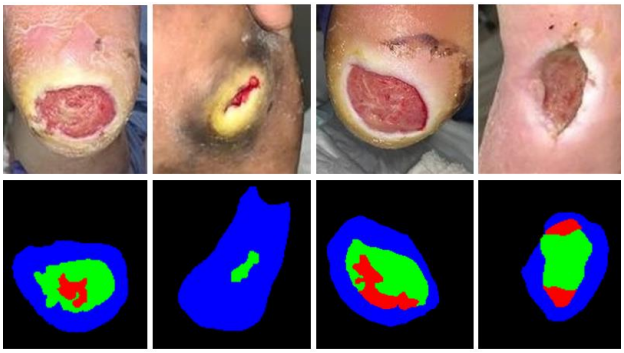


Fig. 1. Diabetic foot ulcer samples from the Chronic Wound (top) and DFUTissue (bottom) datasets. The colors red, green, and blue correspond to fibrin, granulation, and callus, respectively. Image adapted from [15].

We follow a rigorous selection protocol to ensure the quality of the dataset. Images exhibiting any of the following issues were excluded: blurriness, poor lighting, deformation, obstruction by foreign objects, duplicate images or incorrect segmentation. After filtering out invalid images, a wound care specialist reviewed the remaining data to remove any mask that did not adhere to the specified guidelines, further enhancing the dataset's credibility and robustness. In the end, 1,356 images were retained for training and evaluation, which could be considered small for some CNNs. At the outset of this study, the images are randomly partitioned into three subsets: 60% for training, 20% for validation, and 20% for testing.

III. Baseline Model

We introduce the EfficientNetV2-inspired U-Net architecture, adapted as the baseline model for our study.

A. Modeling

EfficientNetV2 was selected for its proven efficiency in fast convergence and strong performance on benchmarks such as ImageNet. Although originally designed for classification tasks [16], we demonstrate that EfficientNetV2 can be effectively repurposed for semantic segmentation by integrating it into a U-Net-style architecture tailored to our domain-specific dataset. Leveraging pre-trained weights, we fine-tune the model as detailed in Section III-B, benefiting from the rich spatial representations already learned. Among the available encoder variants, Model S was selected[2]. The image processing pipeline of our proposed adaptation, along with the number of channels and spatial dimensions at the output of each block, is illustrated in Fig. 2.

The encoder in EfficientNetV2-S has 20.33 million trainable parameters and follows a pattern of eight blocks with increasing complexity. The first block extracts low-level features, while blocks 2–4 refine these features using a series of fused multi-branch convolutional blocks. Blocks 5–7 employ depthwise separable convolutions and squeeze-and-excitation blocks, enhancing deep feature extraction by selectively retaining key activations. The final block outputs to a Global Average Pooling (GAP) layer and Fully Connected Layers (FCL), which are replaced by our decoder. EfficientNetV2 also incorporates stochastic depth for better generalization.

[2]After practical experiments, Model S yielded results similar to Model M and Model L. Given that Model S has fewer parameters, it is faster to train and lighter, making it our model of choice. The other variants were not tested, as their number of parameters was considered insufficient for our task.

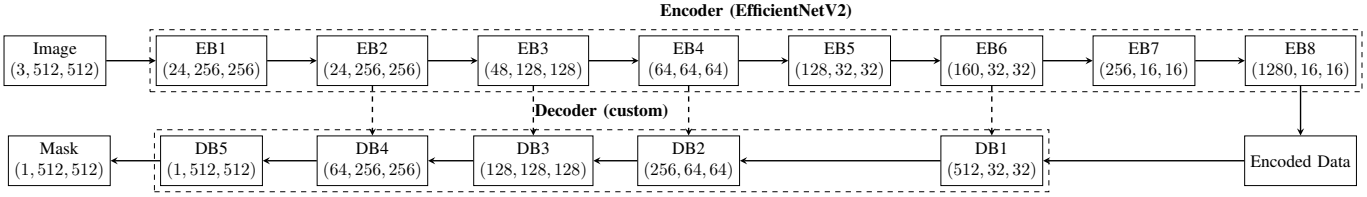


Fig. 2. EfficientNetV2-S image processing pipeline with an example of a 512x512 input image; EB: Encoder Block; DB: Decoder Block.

The decoder follows a U-Net-like structure, replacing the GAP and FCL layers. Each upsampling step doubles the spatial dimensions and halves the channel count, except in the first and last decoder blocks, which adapt the number of channels. After upsampling, a corresponding encoder feature map with matching spatial dimensions is concatenated. A convolutional layer processes these features before the next upsampling step, preserving fine-grained spatial details and enhancing feature propagation. Feature maps from encoder blocks 6, 4, 3, and 2 were selected for their high channel count and spatial relevance. While we tested attention mechanisms for refining skip connections, specifically using a gating mechanism with element-wise multiplication [17], they did not significantly improve performance.

B. Training Setup

Our dataset consists of 512×512 pixel images, which our computational resources can process at full resolution. With a 24GB VRAM limit, using high-resolution images constrains the maximum feasible batch size. Through experimentation, comparing different resolutions and batch sizes showed that preserving full resolution yields better results than increasing batch size at the cost of image quality. To balance stability and efficiency, we use a batch size of 19. We apply transformations from the Albumentations library[3] to triple the training dataset size by applying color manipulation, Gaussian noise and blur, random rotations, shifts, and flips. We adopt the Ranger21 optimizer [18] for its smooth training dynamics, fast convergence, and high accuracy, which were validated on our dataset, where it consistently outperformed alternative optimizers. We set the initial learning rate to 10^{-3} and Weight Decay to 10^{-4}. To permit a fair comparison between the validation and training datasets, we avoid any interventions that could influence the model's training, such as using a learning rate scheduler based on the validation dataset.

We employ the Dice loss function [19], D_L, for training, as it helps mitigate overfitting by emphasizing foreground regions and maintaining gradient flow despite class imbalance. The Dice loss is computed as:

$$D_L = \frac{FP + FN}{2TP + FP + FN + \epsilon} \tag{1}$$

where TP, FP, and FN denote True Positives, False Positives, and False Negatives, respectively, and ϵ is a small constant, set to 10^{-6}, to prevent division by zero.

IV. Experimental Results

We first train the model for 100 epochs to gain insights into its training and validation behavior. Fig. 3 illustrates stable learning up to 20 to 30 epochs, during which the validation curve closely aligns with the training curve, followed by the emergence of overfitting signs. This suggests that training beyond this point does not provide meaningful improvements and may degrade the model's generalization ability.

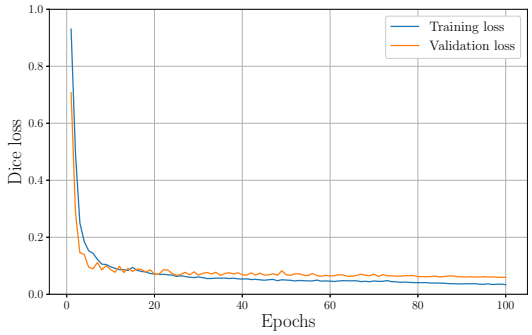


Fig. 3. Training and validation dice loss across epochs.

Later, we compare the number of poorly segmented images between the validation and test datasets. Then, we evaluate the best-performing model across different performance metrics and epoch counts. Using k-fold cross-validation, we compare the computed Dice score with the performance metric across folds. Following this, we manually examine each generated mask to further support and validate the results.

A. Edge cases between training and validation

The validation dataset is crucial for assessing a model's performance on the test dataset while training, and in our experiment, the average Dice score between validation and test dataset remained consistently bounded. However, when analyzing edge cases to evaluate the model's ability to generalize, a weak correlation between validation and test segmentation evolution is observed.

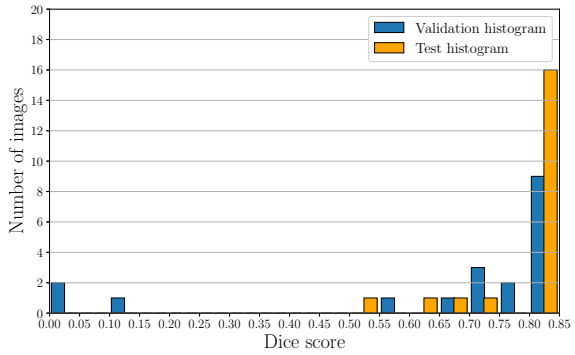
Fig. 4(a) presents a histogram of Dice scores below 85%, to help visualize poorly segmented cases, for the model after 50 epochs of training. It reveals that at least three validation images were segmented badly (Dice score below 15%), while the test dataset maintained a minimum Dice score of 50%. Fig. 4(b) displays the results for the model after an additional epoch of training. While performance improved on the validation dataset, the test dataset exhibited two cases of undetected segmentation (Dice score below 5%).

These observations indicate that while the mean Dice score remains closely correlated across datasets, improvements in generalization may benefit certain cases while negatively impacting others. Consequently, we avoid further experiments at this level, such as selecting models based on the highest minimum Dice score observed in the validation dataset.

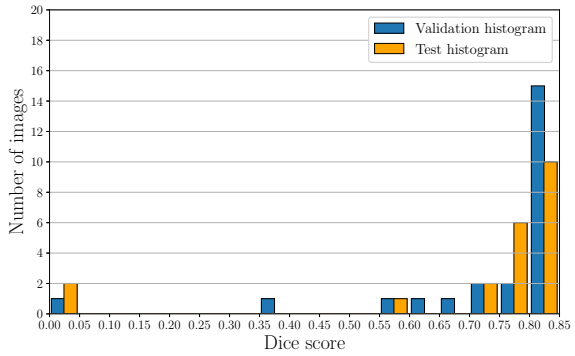
B. Cross-validation

The experiments in the previous subsection demonstrated that, past a certain point, the validation dataset offers little in guiding generalization for our model, as improvements in some cases might come at the expense of others. Also, while it can be useful for identifying the point before overfitting, in our case, performances on the validation dataset fluctuates slightly

[3]Albumentations is an efficient Python library for image augmentation, widely used in computer vision tasks. It offers a variety of transformations and integrates well with deep learning frameworks like TensorFlow and PyTorch. More information is available at: https://explore.albumentations.ai/.



(a) Histogram after 50 epochs.



(b) Histogram after 51 epochs.

Fig. 4. Comparison of Dice scores on edge cases for validation and test datasets at two different epochs.

after 30 epochs (Fig. 3), leading to the potential selection of a model with a marginally higher Dice score after overfitting, despite poorer generalization. Since the best validation results were often recorded towards the end of the training, using a validation set helps little with delimitation of overfitting.

To investigate this, we evaluate two dataset partitioning strategies: the original 60% training, 20% validation, and 20% testing split, and an alternative approach that combines the validation set with the training set in hopes of improving generalization. These experiments were conducted over 100 epochs to evaluate whether using early stopping with a validation dataset to prevent overfitting compensates for the samples excluded from training. Additionally, we evaluate an 80% training split over 30 epochs to prevent overfitting. To ensure a comprehensive analysis, we employ five-fold cross-validation, generating predictions for each image in the dataset.

Table I compares the achieved test Dice scores across folds based on the selected evaluation metric: the lowest average Dice loss for validation or training. Despite having negligible impact on results, the standard deviation (STD) is reported for comparison across models. The model selected using the validation dataset exhibits a higher mean Dice loss. Although it is expected considering we prevent overfit and allocate less resources for training, it brings a lower Dice score on the test dataset. In contrast, incorporating the validation set into training led to improvements across all measured parameters. However, the most striking result is the comparison between training over 30 versus 100 epochs. While the mean Dice loss is higher at 30 epochs, the model achieved a slightly better Dice score on the test dataset.

TABLE I
EVALUATION METRIC COMPARISON.

Evaluation metric	**Mean Dice loss**	**Test Dice score**
Validation (100 epochs)	$6.06\% \pm 0.489$	$93.05\% \pm 0.747$
Training (100 epochs)	$3.56\% \pm 0.049$	$93.58\% \pm 0.573$
Training (30 epochs)	$4.62\% \pm 0.014$	$93.72\% \pm 0.566$

The outputs were then analyzed case by case, providing qualitative assessments of each model's strengths and weaknesses across 1,356 segmentations. Each segmentation was assigned one of three labels:

- "Good" segmentations are fully exploitable, even if minor differences exist with the real mask (e.g., inclusion of other wound-like areas or ambiguous zones such as blurry/dark regions).
- "Uncertain" segmentations are mostly exploitable but contain unexpected minor discrepancies, such as additional understandable false positives (e.g., red nails mistaken for injuries).
- "Bad" segmentations do not capture the full segmentation or fail to identify key features altogether, often resulting in illogical or nonsensical outputs that do not align with the expected mask.

Observations from the case-by-case analysis (Table II) confirm that training the model for fewer epochs resulted in better generalization. Additionally, incorporating the validation dataset into training improved performance on many examples, reducing the number of "Bad" segmentations by half.

TABLE II
MANUAL OBSERVATION RESULTS.

Observation	**Studied metric (epoch)**		
	Val (100)	***Train (100)***	***Train (30)***
Good	1258	1268	1300
Uncertain	68	73	44
Bad	30	15	12

Among the models tested, the training-based model at 30 epochs provided the most reliable segmentations, with logical delineations and minimal unexpected failures. Using the training dataset only for 100 epochs resulted in more logical segmentations than the model based on the validation dataset, but still showed signs of overfitting, with segmentation inconsistencies that were difficult to explain. The validation-based model performed reasonably well, but exhibited notable inconsistencies, especially in overlapping mask regions.

Each of these models demonstrated both strengths and weaknesses across various common cases. However, they exhibited specialization in certain types of injuries—one model performed better on toe wounds, while another excelled in

abdominal wound segmentation. This suggests that, rather than designing a single model to generalize across all wound types, better performance might be achieved by training specialized models for different tissue types.

When comparing our results to those of other studies, it is important to consider differences in model design, dataset relevance, and evaluation protocols. Some works, referenced in [5], report near-perfect Dice scores (around 99%) based on limited and non-representative datasets, which are often selected for visual demonstration purposes rather than for assessing generalization capability. Others, like [20], present inconsistent metrics—e.g., an IoU of 99.9% with a Dice score of 93.4%—which is mathematically incompatible, suggesting flaws in either methodology or reporting. Such studies lack the rigor required for a valid comparison.

TABLE III
ACHIEVED DICE SCORE COMPARISON.

Images	Class	Model	Dice score (%)	Paper
1109	DFU	MobileNetV2+CCL	90.47	[6]
1200	Burn	LinkNet-EffB1	91.70	[21]
2372	Diverse	LinkNet-EffB1	92.09	[7]
1356	Diverse	Ours	**93.72**	Ours

Table III compares our results with selected studies applying sufficiently rigorous and transparent methodologies. Our model achieves a higher Dice score using a diverse dataset, suggesting strong generalization and applicability across various classes.

V. DISCUSSION OF FINDINGS

Our experiments revealed that relying on early stopping with the validation dataset may not yield the most effective model for generalization, particularly when dealing with a dataset considered small, as well as when the validation curve converges to its minimum, as shown in Fig. 3. Integrating the validation set into the training process, as shown in subsection IV-B, led to improved performance.

Additionally, generalizing across diverse injury types led to inconsistent results. The model struggled with conflicting features such as tissue type, shape, and texture (Fig. 4), suggesting that a single model may converge to suboptimal local minima. Training multiple specialized models on narrower injury categories could yield better performance and reduce stagnation.

VI. CONCLUSION

We demonstrated that EfficientNetV2—originally designed for image classification—can serve effectively as an encoder in a U-Net-like architecture for medical image segmentation. Combined with a curated, high-quality dataset and the Ranger21 optimizer, our approach achieved fast convergence and strong segmentation performance, reaching a state-of-the-art Dice score of 94%. Cross-validation and case-by-case analysis provided insights into model behavior, questioning the exclusive use of Dice loss during training. We also examined the limited impact of a separate validation set on overfitting prevention.

REFERENCES

[1] S. A. Eming, P. Martin, and M. Tomic-Canic, "Wound repair and regeneration: mechanisms, signaling, and translation," *Science Translational Medicine*, vol. 6, no. 265, p. 265sr6, Dec. 2014. doi: 10.1126/scitranslmed.3009337.

[2] H. A. Wallace, B. M. Basehore, and P. M. Zito, "Wound healing phases," in *StatPearls*. StatPearls Publishing, Treasure Island, FL, USA, last updated June 12, 2023. [Online]. Available: https://www.ncbi.nlm.nih.gov/books/NBK470443/. [Accessed: Dec. 21, 2025].

[3] R. G. Frykberg and J. Banks, "Challenges in the treatment of chronic wounds," *Advances in Wound Care (New Rochelle)*, vol. 4, no. 9, pp. 560-582, Sep. 2015. doi: 10.1089/wound.2015.0635.

[4] Eastern Connecticut Health Network, "Skin wounds: Types, risks, and treatment options," *ECHN Blog*, [Online]. Available: https://www.echn.org/blog/skin-wounds-types-risks-and-treatment-options/. [Accessed: Dec. 21, 2025].

[5] R. Zhang, D. Tian, D. Xu, W. Qian, and Y. Yao, "A survey of wound image analysis using deep learning: classification, detection, and segmentation," *IEEE Access*, vol. 10, pp. 79502-79515, 2022. doi: 10.1109/ACCESS.2022.3194529.

[6] C. Wang et al., "Fully automatic wound segmentation with deep convolutional neural networks," *Scientific Reports*, vol. 10, article no. 21897, Dec. 2020. doi: 10.1038/s41598-020-78799-w.

[7] A. Mahbod, G. Schaefer, R. Ecker and I. Ellinger, "Automatic Foot Ulcer Segmentation Using an Ensemble of Convolutional Neural Networks," in 2022 26th International Conference on Pattern Recognition (ICPR), Montreal, QC, Canada, 2022, pp. 4358-4364, doi: 10.1109/ICPR56361.2022.9956253.

[8] F. Maleki et al., "Generalizability of machine learning models: quantitative evaluation of three methodological pitfalls," *Radiology: Artificial Intelligence*, Nov. 16, 2022. doi: 10.1148/ryai.220028.

[9] M. Tan and Q. V. Le, "EfficientNetV2: smaller models and faster training," *arXiv preprint*, arXiv:2104.00298, Jun. 2021. [Online].

[10] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "ImageNet-21K Pretraining for the Masses," arXiv preprint arXiv:2104.08724, 2021. [Online]. Available: https://arxiv.org/abs/2104.08724.

[11] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," IJCAI'95: Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2, 1995, pp. 1137-1143.

[12] L. Leoscode, "Wound images segmentation [2760 samples]: Medetec + FUSeg + WSNET with refined annotations," Kaggle, [Online]. Available: https://www.kaggle.com/datasets/leoscode/wound-segmentation-images. [Accessed: Dec. 5, 2024].

[13] S. Thomas, "Stock pictures of wounds," *Medetec Wound Database*, 2020. [Online]. Available: http://www.medetec.co.uk/files/medetec-image-databases.html. [Accessed: Dec. 5, 2024].

[14] S. R. Oota, V. Rowtula, S. Mohammed, M. Liu, and M. Gupta, "WSNet: towards an effective method for wound image segmentation," *Proc. IEEE/CVF Winter Conf. Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, 2023, pp. 3233-3242. doi: 10.1109/WACV56688.2023.00325.

[15] M. K. Dhar et al., "Wound tissue segmentation in diabetic foot ulcer images using deep learning: a pilot study," *arXiv preprint*, arXiv:2406.16012, Jun. 2024.

[16] V. Venugopal, N. I. Raj, M. K. Nath, and N. Stephen, "A deep neural network using modified EfficientNet for skin cancer detection in dermoscopic images," *Decision Analytics Journal*, vol. 8, article no. 100278, 2023. doi: 10.1016/j.dajour.2023.100278.

[17] O. Oktay et al., "Attention U-Net: Learning where to look for the pancreas," arXiv. https://doi.org/10.48550/arXiv.1804.03999.

[18] L. Wright and N. Demeure, "Ranger21: a synergistic deep learning optimizer," *arXiv preprint*, arXiv:2106.13731, Aug. 2021.

[19] R. Azad et al, "Loss functions in the era of semantic segmentation: A survey and outlook," arXiv preprint arXiv:2312.05391, Dec. 2023. [Online]. Available: https://arxiv.org/abs/2312.05391

[20] J. Chae, K. Y. Hong, and J. Kim, "A pressure ulcer care system for remote medical assistance: Residual U-Net with an attention model based for wound area segmentation," *arXiv preprint*, arXiv:2101.09433, Apr. 2021. [Online]. Available: https://arxiv.org/abs/2101.09433

[21] H. Liu, K. Yue, S. Cheng, W. Li, and Z. Fu, "A framework for automatic burn image segmentation and burn depth diagnosis using deep learning," *Computational Intelligence for Health Care*, Special Issue, 2023.