Fast Semantic Segmentation of Medical Images

António Carvalho
ISEL/IPL
Lisboa, Portugal
a48347@alunos.isel.pt

Mário Vestias
INESC INOV/ISEL/IPL
Lisboa, Portugal
mario.vestias@isel.pt

***Abstract*—Semantic segmentation of medical images enables automatic identification and localization of anatomical structures and pathological regions. The technique classifies image regions at the pixel level, offering a powerful method for accurate diagnosis of diseases. Despite many advancements in deep learning models with high accuracy, the deployment of these systems in real-world clinical settings is slowed down by the computational requirements of these models, leading to inefficient systems with slow processing times. The work proposed in this paper is an efficient and fast semantic segmentation network tailored for deployment on an embedded computing system. By optimizing the model to reduce its size and computational complexity and applying hardware-oriented optimizations, this work aims to overcome current limitations in computational efficiency and latency. The results show a reduction of more than $10\times$ in model size compared to state-of-the-art works with a similar accuracy.**

***Index Terms*—Semantic Segmentation, Image Processing, Embedded Computing, Deep Learning**

I. Introduction

In the last decade, computational resources have improved exponentially, alongside the massive availability of datasets, leading to a rapid growth in the artificial intelligence (AI) field. This technological progress has revolutionized many sectors, with healthcare being one of the most significant to experience this transformation. AI-driven approaches have demonstrated significant success in automating and enhancing medical analysis, and are even capable of outperforming human experts in certain tasks.

Today AI is used in healthcare for applications like disease detection, prognosis prediction, automated image interpretation, and many other [1]. This work focuses on fast feature extraction from medical images through semantic segmentation and deep learning algorithms.

Semantic segmentation is a technique utilized for classifying image regions at the pixel level, allowing for the distinction of elements in the background from elements in the foreground. This feature extraction is very important in the medical sector as it provides aided diagnoses and allows for automated medical imaging analysis. For these image classification and pattern recognition tasks, convolutional neural networks (CNNs) [2] have become the standard approach. They are commonly used for tasks where the output is a single class label, such as car, plane, bird, etc. However, in the medical segmentation field, the output should include the localization of the object, for example, a tumor, organ delineation, and lesion classification [3], introducing what is known as pixel-wise classification.

In most of the medical image semantic segmentation field, U-Net [4] has emerged as the groundwork. Their proposed *U* shaped architecture became the preferred model for medical image segmentation. As a result of this architecture, a lot of meaningful improvements have been proposed, leading to tremendous success in several medical applications such as cardiac segmentation from Magnetic Resonance Imaging (MRI) [5] and prostate segmentation from MRI [6].

While semantic segmentation networks continue to achieve impressive results, they are also becoming larger and less computationally efficient, posing challenges for real-time applications, especially in resource-constrained environments [7].

In this work, the current state-of-the-art in medical image semantic segmentation is analyzed to identify and enhance an existing model, reducing its complexity to be deployed in an embedded system while maintaining high performance. From this research, a very lightweight (40K params) and optimized network based on the CMUNeXt [8] network is proposed.

II. Related Work

Many deep learning-based image segmentation models have been proposed. These models vary in dimension, encoding, and many other aspects. To understand the context behind these models, what networks have been developed, and what improvements or enhancements result from them.

A. Encoder-Decoder & Semantic Segmentation

The encoder-decoder architecture is the most common deep learning architecture for semantic segmentation. It consists of an encoder and a decoder. The encoder consists of a series of deep convolutional layers combined with downsampling operations, designed to extract high-level features from the input image. These features aim to capture pixel-level semantic information about objects, edges, and background regions. The decoder then processes these features using upsampling and/or deconvolution layers to reconstruct the final segmentation mask. The mask represents the probability of pixels belonging to the background or to the foreground. Many implementations of this deep learning network have been proposed [9], [10].Following the same architecture, the great breakthrough in medical image semantic segmentation came from U-Net [4]. Thanks to their base work, many meaningful improvements to the *U* shaped network have been proposed.

B. Vision Transformers

Originally developed for natural language processing (NLP), transformers have since been successfully adapted for a wide range of computer vision tasks. These attention-based architectures have become very popular in the computer vision field by enabling global context extraction. The Vision Transformer (ViT) [11] introduced an alternative based on self-attention layers for sequence-to-sequence prediction, achieving state-of-the-art performance.

In the context of medical image semantic segmentation, several transformer-enhanced variants of the original U-Net architecture have appeared, integrating self-attention mechanisms to enhance performance and precision. TransUnet [12] combines CNNs with transformers, Swin-UNet [13] leverages hierarchical Swin Transformer blocks,

C. Multi-Layer Perceptron Module

ViT-based architectures mostly focus on improving network performance and precision but often overlook aspects like computational complexity, inference time, and model size. These aspects are important for real-world applications as they impact computational efficiency. The Multi-Layer Perceptron (MLP) blocks have emerged as an alternative to the self-attention mechanisms proposed in transformer-based architectures, by combining efficient feature extraction with reduced computational effort.

The MLP-Mixer [14] introduced a conceptually and technically simpler architecture that replaces convolutions and self-attention with token-mixing and channel-mixing layers based exclusively on MLPs. This work demonstrated that competitive performance could be obtained without relying on self-attention mechanisms or convolutions inspiring the development of multiple MLP-based architectures for semantic segmentation. Among these, AS-MLP [15], optimized spatial mixing by shifting features along axial directions, improving efficiency while maintaining strong spatial awareness, ResMLP [16] incorporated residual connections within MLP layers to improve gradient flow and training stability, while S^2-MLP [17] enhanced spatial information propagation through spatial-shift operations, significantly reducing computational costs.

In medical image semantic segmentation, these MLP-Mixer blocks were used in UNeXt [18], a very recent work that successfully integrates the MLP-Mixer with a U-Net-based network, reducing significantly the number of parameters and the computational complexity. This mixer is also present in CMU-Net [19] by mixing features at distant spatial locations.

D. Depthwise Separable Convolutions

Depthwise separable convolutions are designed to enhance computational efficiency and reduce the number of parameters in deep learning models. This design separates a convolution into two operations: a depthwise convolution, which applies a single convolutional filter per input channel, followed by a pointwise convolution, which uses a 1×1 convolution to combine the outputs of the depthwise convolution across channels. This reduces the computational complexity of the convolutional layer compared to the standard 3D convolutional layer.

This design was popularized by MobileNetV2 [20] and proved to be very effective for lightweight deep neural networks. MobileNetV2 achieved state-of-the-art performance with minimal loss in accuracy, being suited for mobile devices with restrained resources.

E. Lightweight Networks for Medical Imaging

Numerous networks have been proposed to enable compact and efficient medical image segmentation, aiming to handle complex tasks while minimizing the number of parameters and computational overhead.

For multimodal biomedical image segmentation CFPNet-M [21], a lightweight encoder-decoder-based network specifically developed for real-time segmentation using feature pyramid channels.

Dinh et al. [22] demonstrated that U-Lite, a CNN-based model with only one million parameters, is sufficient for medical image segmentation. Al-Fahsi et al. [23] proposed GIVTED-Net, a network combining GhostNet [24], Involution [25], and ViT for lightweight medical image segmentation and She et al. [26] introduced LUCF-Net, a lightweight U-shaped cascade fusion network, optimizing multi-scale feature extraction.

Lastly Tang et al. [8] introduced CMUNeXt, leveraging depthwise separable convolutions from MobileNetV2 to enhance performance, integrating concepts from ConvUNext [27] and ConvMixer [19] to combine the structural advantages of large kernel convolutions with depthwise separability.

III. Fast Medical Image Segmentation Model

A. Architecture

After a thorough analysis of the lightweight models explored in Section II-E, it was decided to work on the CMUNeXt architecture proposed by Tang et al. This architecture is very modular, comprehensive, and lightweight while still achieving state-of-the-art results, making it the perfect candidate for this project. The model proposed in this work is very close to the original, but with some considerable changes, namely the activation function and the removal of the concatenation operations. Also, the final model is considerably smaller compared to the smaller variant of the original CMUNeXt network (CMUNeXt-S).

The overall architecture of the proposed network consists of five layers (L) divided into two stages: the encoder and the decoder stages (See Figure 1).

The encoder is where the proposed model extracts global context information while also being computationally efficient, followed by a simple 2D convolution block and a downsampling operation using Maxpooling. In the decoder stage, the fusion block fuses the semantic features identified by the encoder with the upsampled features from the decoder, summing the skip connections with the previous upsampled layer. Just like the original network, the layers are denoted

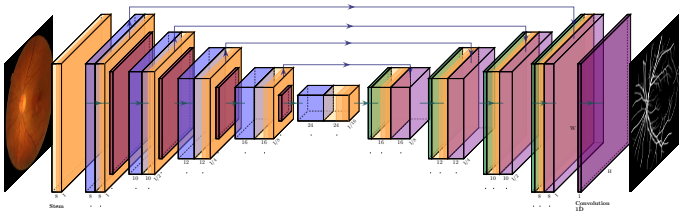

Fig. 1: Outline of the proposed CNN model

from top to bottom and from *L1* to *L5*, also the kernel sizes of each block are denoted from *K1* to *K5* and the number of channels from *C1* to *C5*, this information is relevant for structuring in future references.

B. Activation Function

Another difference in the proposed network is the use of the Hardswish activation function, popularized in the MobileNetV3 [28]. This function tries to emulate the GELU activation function, used in the original network, but with a computationally less expensive. While GELU relies on the error function to approximate a smooth transition, Hardswish replaces it with a simple piecewise linear function. This makes Hardswish more efficient while still retaining non-linearity benefits similar to GELU.

The Hardswish activation function is defined as:

$$\text{Hardswish}(x) = \begin{cases} 0, & \text{if } x \leq -3, \\ x, & \text{if } x \geq +3, \\ x \cdot (x+3) \cdot \frac{1}{6}, & \text{otherwise.} \end{cases}$$

In comparison, the GELU activation function is given by:

$$\text{GELU}(x) = x \cdot \Phi(x)$$

where $\Phi(x)$ is the cumulative distribution function of the standard normal distribution.

Figure 2 illustrates the comparison between the Hardswish and GELU activation functions. The blue curve represents the Hardswish function, while the dashed red curve represents the GELU function. As seen in the plot, Hardswish approximates the GELU function's smooth curve, but with a simpler piecewise linear form.

Throughout the project, multiple activation functions were tested, namely: ReLU, GELU, LeakyReLU, and Hardswish. The differences in accuracy are marginal and, therefore, ReLU was adopted.

C. Encoder

The stem is designed to extract features from the original image at the top level, it is composed of a 2D Convolution layer with a kernel 3×3 , a stride of 1, and a padding of 1, followed by a 2D Batch Normalization layer and an inplace Hardswish Activation layer.

Just like the original network, this block is the most important component of the network and is characterized by the

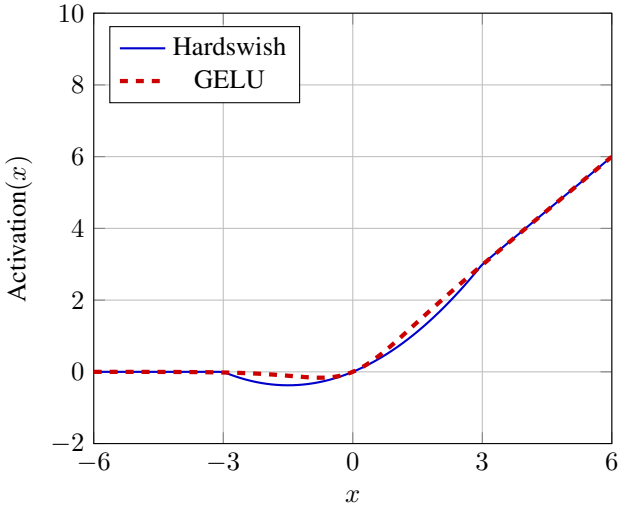

Fig. 2: Comparison of Hardswish and GELU activation functions.

use of Depthwise Separable Convolutions. This is where most of the feature extraction occurs, and therefore it is a crucial component in the network. It consists of two Depthwise 2D Convolution Layers connected sequentially with a variable kernel size of $k \times k$, group size of dim dimension, and a padding of $k \div k$. This is followed by an expanding Pointwise 2D Convolution that transforms the feature map from dim to $dim \times 4$, using a fixed kernel size of 1×1, a Hardswish activation function, and a 2D Batch Normalization layer. Next, a contracting Pointwise 2D Convolution that reduces the feature map from $dim \times 4$ to dim, again with a fixed kernel size of 1×1 and the same activation and normalization functions all connected sequentially. This block is repeated *L* times, with each replication attempting to enhance the network feature extraction capacity, but at the cost of increasing the network parameters and size.

The downsampling operation is performed using a 2D MaxPooling function with a fixed kernel size of 2×2 and a stride of 2, reducing the resolution of the feature map by half.

D. Decoder

1) Skip Fusion: The Skip Fusion block fuses semantic features extracted from the encoder with up-sampled features generated by the decoder. It is composed of a group convolutional layers as the main layer. The convolution operation in the Skip Fusion block is divided into two groups, each extractingfeatures independently on the encoder's connections and the upsampled decoder features. This group convolution uses a kernel size of 3×3, a stride of 1, and padding of 1, followed by two pointwise (1×1) convolutions.

Each convolutional layer within the Skip Fusion block is followed by a Hardswish activation function and a 2D Batch Normalization layer to enhance training stability and learning efficiency. The Skip Fusion block is defined as:

$$f_{concat} = Sum\left(\begin{array}{c} BN\{Conv2D(f_{\mathcal{E}})\}, \\ BN\{Conv2D(f_{\mathcal{D}})\} \end{array} \right) \quad (1)$$

$$f'_{fusion} = BN\left(\sigma_1\left\{PointwiseConv2D\left(f_{concat}\right)\right\}\right) \quad (2)$$

$$f_{fusion} = BN\left(\sigma_1\left\{PointwiseConv2D\left(f'_{fusion}\right)\right\}\right) \quad (3)$$

where f_{fusion} is the final fused feature map, and f_E and f_D are the encoder and decoder features, respectively.

The upsampling block includes an upsampling, a convolutional, and batch normalization layers, followed by a Hardswish activation function. The upsample layer uses bilinear interpolation with a factor of two.

IV. Experiments

This section provides the results of the proposed image segmentation model against several state-of-the-art image segmentation networks, describing the datasets used in our experiments, followed by the training protocols, evaluation metrics, and a discussion of the various network models.

A. Datasets

Two datasets were considered in this work:

a) BUSI: The Breast UltraSound Images (BUSI) [29] dataset includes 780 breast ultrasound images, including 133 normal cases, 487 benign cases, and 210 malignant cases.From these sets, only the benign and malignant sets were utilized.

b) ISIC2016: The ISIC 2016 [30] dataset consists of dermoscopic images for skin lesion classification. It includes 900 training images and 379 test. Each image is provided with ground truth segmentation masks and diagnostic labels.

B. Training

All networks were trained in PyTorch using CUDA version 11.8 with the same conditions for a total of **300 epochs** and a **batch size** of 8. All input images were resized to 256×256 pixels and normalized. The **Adam optimizer** with a **learning rate** of 1×10^{-3} and **weight decay of** 1×10^{-4} was used for optimization. All the experiments were conducted using a single NVIDIA GeForce RTX4080 GPU.

1) Evaluation Metrics: All models were evaluated using the following key metrics:

- **IoU (Intersection over Union)**: IoU quantifies the overlap between the predicted segmentation mask and the ground truth mask:

$$\text{IoU} = \frac{|A \cap B|}{|A \cup B|}$$

where A and B are the predicted and ground truth regions, respectively.

- **PC (Precision)**: Precision evaluates the model's ability to correctly identify positive samples among all instances predicted as positive:

$$\text{PC} = \frac{TP}{TP + FP}$$

where TP is the number of true positives and FP is the number of false positives.

- **F1 Score**: F1 Score is the harmonic mean of precision and recall, offering a balanced evaluation of a model's performance, useful in imbalanced class distributions. It combines both precision and recall into a single metric, emphasizing their trade-off:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- **AAC (Average Accuracy per Class)**: The Average Accuracy per Class (AAC) measures the average accuracy for each class, as follows:

$$\text{AAC} = \frac{1}{N} \sum_{i=1}^{N} \frac{TP_i}{TP_i + FN_i}$$

where N is the number of classes, and TP_i and FN_i are the true positives and false negatives for class i, respectively.

- **Loss**: The loss function $\mathcal{L}$ quantifies the difference between the predicted output $\hat{y}$ and the ground truth y. It is calculated as a weighted combination of binary cross-entropy (BCE) and dice loss (Dice):

$$\mathcal{L} = 0.5 \cdot \text{BCE}(\hat{y}, y) + \text{Dice}(\hat{y}, y) \quad (4)$$

2) Network Models: Multiple neural networks ranging from lightweight models to large models and transformer-based models were considered in our experiments (See Table I).

TABLE I: Networks used in our experiments.

Category	Network
U-Net Variants	UNet UNetV2 UNet++
Lightweight	CMUNeXt CFPNetM UNeXt GIVTED-Net LUCF-Net
Transformer-based	TransUnet
Proposed Network	Mobile-CMUNeXt

C. CMUNeXt-XXS

The **CMUNeXt-XXS** builds upon the CMUNeXt family of architectures with optimizations to enhance feature extraction, reduce parameter count, and improve inference speed. Compared to the original model and its variants, CMUNeXt-XXS has a significant reduction in the number of channels and block lengths but keeps the same large kernels as the CMUNeXt-S. Table II illustrates a comparison between different CMUNeXt variants and the proposed CMUNeXt-XXS model.

D. Results

The training results can be found in Table III. All metrics are reported with two decimal places.

Given the results, CMUNeXt-XXS demonstrates an exceptional balance between efficiency and performance across all

TABLE II: CMUNeXt variants and CMUNeXt-XXS

Network	Number of Channels					Length of Blocks					Kernel Size				
	$C1$	$C2$	$C3$	$C4$	$C5$	$L1$	$L2$	$L3$	$L4$	$L5$	$K1$	$K2$	$K3$	$K4$	$K5$
CMUNeXt-L	32	64	128	256	512	1	1	1	6	3	3	3	7	7	7
CMUNeXt	16	32	128	160	256	1	1	1	3	1	3	3	7	7	7
CMUNeXt-S	8	16	32	64	128	1	1	1	1	1	3	3	7	7	9
CMUNeXt-XXS	8	10	12	16	24	3	1	1	2	3	3	3	7	7	9

TABLE III: Results on Medical Datasets (ISIC2016, BUSI)

Network	Params (M)↓	MACS (G)↓	Metrics (%)					
			ISIC2016			BUSI		
			IoU↑	F1↑	AAC↑	IoU↑	F1↑	AAC↑
U-Net	34.52	65.52	83.00	90.59	95.36	61.10	74.89	95.37
U-Net++	9.16	34.90	83.67	90.88	85.57	61.16	73.88	95.66
UNetV2	24.90	5.10	83.73	90.93	95.50	63.28	76.37	95.76
TransUnet	105.32	38.52	84.70	91.56	95.84	65.20	77.81	95.45
UNeXt-S	0.25	**0.10**	84.12	91.26	95.61	60.59	74.82	95.23
GIVTED-Net	0.19	0.37	**85.91**	**92.34**	**96.14**	64.60	77.92	95.86
LUCF-Net	6.93	8.59	85.37	92.04	96.04	66.17	79.10	96.19
CFPNetM	0.76	3.47	85.16	91.88	96.04	**67.10**	**79.76**	**96.33**
CMUNeXt-S	0.42	1.09	84.87	91.71	95.87	64.12	77.39	95.84
CMUNeXt	3.14	7.41	84.99	91.79	95.93	65.84	78.94	96.31
CMUNeXt-L	8.28	17.18	85.03	91.83	95.83	65.83	78.63	96.02
CMUNeXt-XXS	**0.04**	0.47	84.94	91.75	95.80	65.81	78.93	96.12

datasets. With only **0.04M parameters** and **0.47G MACs**, it is the lightest model in the benchmark while still maintaining competitive segmentation accuracy.

Compared to the base CMUNeXt model (**3.14M parameters, 7.41G MACs**), CMUNeXt-XXS achieves a **98.7% reduction in parameters** and **93.6% reduction in MACs** while maintaining comparable performance across ISIC2016, BUSI datasets.

- **ISIC2016:** CMUNeXt-XXS achieves an **IoU of 84.94%**, very close to CMUNeXt (84.99%) and outperforming CMUNeXt-S (84.87%).
- **BUSI:** CMUNeXt-XXS achieves **65.81% IoU**, outperforming CMUNeXt-S (64.12%) and coming close to CMUNeXt (65.84%). It also performs competitively with CFPNetM (67.10%), which has **19x more parameters**.

Overall, CMUNeXt-XXS proves to be an efficient alternative to the existing models, offering state-of-the-art segmentation performance while being lightweight and fast executing in resource-constrained computing platforms.

V. Conclusions and Future Work

An optimized model for medical image segmentation is proposed, aiming to significantly reduce model size and computational complexity while maintaining comparable accuracy.

Experimental results demonstrate that the proposed model achieves approximately 10× fewer parameters and 50% lower computational complexity compared to the lightweight version of the original architecture.

To further minimize resource usage, the model will undergo quantization and be specifically designed, developed, and deployed on a low-density FPGA platform, ensuring efficient performance in resource-constrained environments.

Acknowledgment

This work was supported by national funds through Fundação para a Ciência e a Tecnologia (FCT) with reference 2023.15325.PEX (OSiRIS), and has also been supported by the project with reference IPL/IDI&CA2024/CSAT-OBC_ISEL through Instituto Politécnico de Lisboa.

REFERENCES

[1] F. Wang and A. Preininger, "Ai in health: state of the art, challenges, and future directions," *Yearbook of medical informatics*, vol. 28, no. 01, pp. 016–026, 2019.

[2] M. Véstias, *Research Anthology on Artificial Neural Network Applications*. IGI GLobal Scientific Publishing, 2022, ch. Convolutional Neural Network.

[3] Y. Fu, Y. Lei, T. Wang, W. J. Curran, T. Liu, and X. Yang, "A review of deep learning based methods for medical image multi-organ segmentation," *Physica Medica*, vol. 85, pp. 107–122, 2021.

[4] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*. Springer, 2015, pp. 234–241.

[5] L. Yu, J.-Z. Cheng, Q. Dou, X. Yang, H. Chen, J. Qin, and P.-A. Heng, "Automatic 3d cardiovascular mr segmentation with densely-connected volumetric convnets," in *Medical Image Computing and Computer-Assisted Intervention- MICCAI 2017: 20th International Conference, Quebec City, QC, Canada, September 11-13, 2017, Proceedings, Part II 20*. Springer, 2017, pp. 287–295.

[6] F. Milletari, N. Navab, and S.-A. Ahmadi, "V-net: Fully convolutional neural networks for volumetric medical image segmentation," in *2016 fourth international conference on 3D vision (3DV)*. Ieee, 2016, pp. 565–571.

[7] M. Véstias, *Processing Systems for Deep Learning Inference on Edge Devices*. Cham: Springer International Publishing, 2020, pp. 213–240.

[8] F. Tang, J. Ding, Q. Quan, L. Wang, C. Ning, and S. K. Zhou, "Cmunext: An efficient medical image segmentation network based on large kernel and skip fusion," in *2024 IEEE International Symposium on Biomedical Imaging (ISBI)*. IEEE, 2024, pp. 1–5.

[9] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[10] V. Badrinarayanan, A. Kendall, and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[11] D. Alexey, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv: 2010.11929*, 2020.

[12] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A. L. Yuille, and Y. Zhou, "Transunet: Transformers make strong encoders for medical image segmentation," *arXiv preprint arXiv:2102.04306*, 2021.

[13] H. Cao, Y. Wang, J. Chen, D. Jiang, X. Zhang, Q. Tian, and M. Wang, "Swin-unet: Unet-like pure transformer for medical image segmentation," in *European conference on computer vision*. Springer, 2022, pp. 205–218.

[14] I. O. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, A. Steiner, D. Keysers, J. Uszkoreit *et al.*, "Mlp-mixer: An all-mlp architecture for vision," pp. 24 261–24 272, 2021.

[15] D. Lian, Z. Yu, X. Sun, and S. Gao, "As-mlp: An axial shifted mlp architecture for vision," *arXiv preprint arXiv:2107.08391*, 2021.

[16] H. Touvron, P. Bojanowski, M. Caron, M. Cord, A. El-Nouby, E. Grave, G. Izacard, A. Joulin, G. Synnaeve, J. Verbeek *et al.*, "Resmlp: Feedforward networks for image classification with data-efficient training," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 4, pp. 5314–5321, 2022.

[17] T. Yu, X. Li, Y. Cai, M. Sun, and P. Li, "S2-mlp: Spatial-shift mlp architecture for vision," in *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, 2022, pp. 297–306.

[18] Z. Chang, M. Xu, Y. Wei, J. Lian, C. Zhang, and C. Li, "Unext: An efficient network for the semantic segmentation of high-resolution remote sensing images," *Sensors*, vol. 24, no. 20, p. 6655, 2024.

[19] F. Tang, L. Wang, C. Ning, M. Xian, and J. Ding, "Cmu-net: a strong convmixer-based medical ultrasound image segmentation network," in *2023 IEEE 20th international symposium on biomedical imaging (ISBI)*. IEEE, 2023, pp. 1–5.

[20] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.

[21] A. Lou, S. Guan, and M. Loew, "Cfpnet-m: A light-weight encoder-decoder based network for multimodal biomedical image real-time segmentation," *Computers in Biology and Medicine*, vol. 154, p. 106579, 2023.

[22] B.-D. Dinh, T.-T. Nguyen, T.-T. Tran, and V.-T. Pham, "1m parameters are enough? a lightweight cnn-based model for medical image segmentation," in *2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2023, pp. 1279–1284.

[23] R. D. H. Al-Fahsi, A. N. F. Prawirosoenoto, H. A. Nugroho, and I. Ardiyanto, "Givted-net: Ghostnet-mobile involution vit encoder-decoder network for lightweight medical image segmentation," *IEEE Access*, vol. 12, pp. 81 281–81 292, 2024.

[24] K. Han, Y. Wang, Q. Tian, J. Guo, C. Xu, and C. Xu, "Ghostnet: More features from cheap operations," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2020, pp. 1580–1589.

[25] D. Li, J. Hu, C. Wang, X. Li, Q. She, L. Zhu, T. Zhang, and Q. Chen, "Involution: Inverting the inherence of convolution for visual recognition," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 12 321–12 330.

[26] Q. She, S. Sun, Y. Ma, R. Li, and Y. Zhang, "Lucf-net: Lightweight u-shaped cascade fusion network for medical image segmentation," *IEEE Journal of Biomedical and Health Informatics*, 2024.

[27] Z. Han, M. Jian, and G.-G. Wang, "Convunext: An efficient convolution neural network for medical image segmentation," *Knowledge-based systems*, vol. 253, p. 109512, 2022.

[28] B. Koonce and B. Koonce, "Mobilenetv3," *Convolutional Neural Networks with Swift for Tensorflow: Image Recognition and Dataset Categorization*, pp. 125–144, 2021.

[29] W. Al-Dhabyani, M. Gomaa, H. Khaled, and A. Fahmy, "Dataset of breast ultrasound images," *Data in brief*, vol. 28, p. 104863, 2020.

[30] ISIC, "Isic challenge," 2016. [Online]. Available: https://challenge.isic-archive.com/landing/2016/