Title of the Project:
Half-UNet: A Simplified U-Net Architecture for Medical Image Segmentation

Team members:
Jai Raj Choudhary | jchoudhary@hawk.iit.edu
Laylo Karimova | lkarimova@hawk.iit.edu
Chao Yang | cyang72@hawk.iit.edu

Abstract
This paper introduces a new model called Half-UNet, which simplifies the popular U-Net model used for cutting out specific areas in medical images. U-Net is known for being complex and requiring a lot of computer power. Half-UNet changes the design to make it less complicated and easier on resources. It does this by streamlining how information is processed and introducing new techniques for combining image features. We tested Half-UNet on different medical images like mammograms, CT scans of lungs, and heart MRI images. The results were impressive: Half-UNet was just as accurate as the original U-Net but needed much less computing power and had far fewer parameters to adjust. This finding challenges the idea that U-Net needs to be complex to work well and shows that a simpler model like Half-UNet can do the job effectively with fewer resources.

1 Introduction
Medical image analysis, particularly medical image segmentation, plays a vital role in modern diagnostic and treatment methodologies. This process is crucial for accurate diagnosis, treatment planning, and monitoring disease progression. The integration of deep learning methods, especially Convolutional Neural Networks (CNNs), has significantly enhanced the accuracy and efficiency of these Analyses.

Current state-of-the-art image segmentation models include fully convolutional networks (FCN) (Long et al., 2015), U-Net (Ronneberger et al., 2015), SegNet (Badrinarayanan et al., 2017), PSPNet (Zhao et al., 2017), and several DeepLab versions (Chen et al., 2017a,b, 2018). These models make use of unique information from various scales. U-Net is among the most commonly employed methods in the medical image segmentation and it is believed that the success of this method depends on its U-shaped structure.

However, previous works like "U-Net: Convolutional Networks for Biomedical Image Segmentation" have established the U-Net framework while highlighting concerns about its complexity and resource demands. Similarly, "U2-Net: Going Deeper with Nested U-Structure for Salient Object Detection" further explored U-Net's potential but faced challenges due to its intricate nested network structures. These works underscore the need for a more efficient architecture.

Numerous models based on the U-Net structure have been proposed since, each offering distinct enhancements and optimizations.

Residual Units U-net
Introduced by Kerfoot et al. (2018), the architecture is built from residual units and aims to enhance the segmentation of the left ventricle.

H-DenseUNet
Li et al. (2018) adapted the U-Net model by using dense convolutions for more effective liver and tumor segmentation tasks.

UNet++
Proposed by Zhou et al. (2018), this model introduces nested and densely con- nected skip pathways, reducing the semantic gap between encoder and decoder components. However, its nested network structure can become overly complex.

NAS-UNet
Weng et al. (2019) presented a novel approach, employing neural architecture search to automatically discover optimal architectures for both downsampling (DownSC) and upsampling (UpSC) paths. This resulted in a significant reduc- tion of parameters while improving performance.

UNet3+
With an aim to leverage all-scale feature maps, Huang et al. (2020) developed UNet3+, which combines features at every level of fusion, enhancing the usage of full-scale feature information. This approach offers competitive results with fewer parameters than the traditional U-Net.

DC-UNet
Proposed by Lou et al. (2021), this version analyzed the classic U-Net and MultiResUNet architectures. It introduces a Dual-Channel CNN block, aiming to provide more effective feature extraction with a reduced parameter count.

Despite the variety and innovation of these U-Net variants, they all rely on to the fundamental U-shaped structure of the original U-Net. Nevertheless, each model attempts to balance complexity, efficiency, and performance, often focusing on reducing the number of parameters and floating-point operations (FLOPs). Considering the limitations of U-Net, specifically its U-shaped structure, which isn't necessarily optimal for segmentation tasks - Half-UNet have been introduced.

Introducing Half-UNet, this paper proposes a simplified and efficient encoder-decoder network based on U-Net, where both the encoder and decoder are streamlined. The architecture innovatively employs channel number unification, full-scale feature fusion, and Ghost modules to reduce complexity. This design addresses the excessive parameter count and computational load of traditional U-Net and its variants.

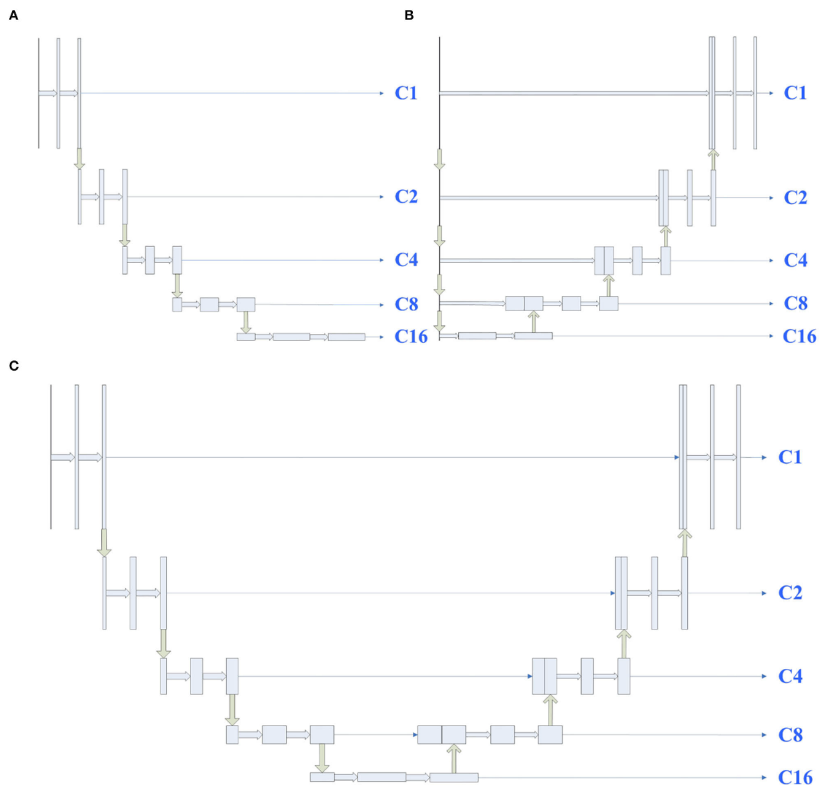
Comparative studies were conducted across various medical image segmentation tasks, including mammography, lung nodule CT, and left ventricular MRI segmentation. The results demonstrate that Half-UNet, despite its simplicity, matches the accuracy of U-Net and its complex variants. Remarkably, it accomplishes this with at least 97.6% fewer parameters and 81.6% fewer floating-point operations than the original U-Net.

This paper not only presents a more resource-efficient model for medical image segmentation but also clarifies misconceptions about the necessity of U-Net's U-shaped structure. By analyzing network structures and performing empirical testing, the paper provides insights into achieving high segmentation performance with significantly reduced computational demands. Half-UNet emerges as a promising solution for effective and efficient medical image analysis.

2 Problem Description

The U-Net framework has gained prominence for its efficacy in medical image segmentation, yet there persists an ongoing debate regarding the necessity and efficiency of its U-shaped, symmetric architecture. In particular, the question arises as to which component of the U-Net's structure is principally responsible for its segmentation success. Investigative work by Chen et al. (2021) provides a pivotal comparison between various encoder strategies, such as Multiple-in-Multiple-out (MiMo), Single-in-Multiple-out (SiMo), Multiple-in-Single-out (MiSo), and Single-in-Single-out (SiSo), challenging the presumed superiority of complex multi-scale feature fusion methods.

These studies reveal that a SiMo encoder configuration could nearly match the performance of the more complex MiMo setup, indicative of the Feature Pyramid Network (FPN) approach. Such results propose that the simpler divide-and-conquer strategy, a method central to the U-Net's encoder that segregates the input image into varying scales for processing, might be more critical than the intricate feature fusion traditionally employed in the U-Net's decoder. This decoder has been characterized by its transformation of multiple feature map scales into a singular comprehensive feature map, a process repeated several times throughout the network.



The structures of encoders **(A–C)** are derived from UNet’s **encoder**, **decoder**, and **full structure** parts respectively.

To further probe the individual impacts of the encoder and decoder components of U-Net, we re-evaluated the architecture by treating U-Net's encoder and decoder separately, and compared them to U-Net's complete structure. Through this experiment, we discovered that the performance of Encoder A was comparable to that of Encoder C. This indicates that the significance of the decoder's complexity might be less critical than previously thought.

Building on these insights, we suggest that simplifying the feature fusion component of U-Net could potentially preserve segmentation accuracy. Such a modification opens the possibility of creating a more streamlined and computationally efficient model, challenging a key issue in the existing U-Net design without sacrificing performance.

3 Network Architecture

The Half-UNet architecture is designed to address the computational inefficiencies found in the traditional U-Net structure. It diverges from the U-Net framework by simplifying the model through several strategic modifications aimed at reducing complexity without sacrificing performance.

Channel Unification: In U-Net's architecture, each downsampling step doubles the feature channels, adding complexity through larger weight matrices and more parameters, which increases memory usage and overfitting risk. Additionally, with more channels, the computational cost in FLOPs (Floating Point Operations) rises due to more operations needed in convolutions, thereby increasing the overall computational load.

In Half-UNet, the architecture diverges from the doubling strategy of U-Net by keeping the number of channels consistent throughout the network, or "unified." This means that instead of increasing the channels with each downsampling step, Half-UNet maintains a set number of channels across all layers. This has several implications:

- ***Parameter Reduction:*** By unifying the channel numbers, the growth of the weight matrix size is controlled, leading to a smaller model with fewer parameters. Fewer parameters mean less memory storage is required, and the network is faster to train. Additionally, this can help prevent overfitting, as there are fewer degrees of freedom for the model to fit to noise in the training data.
- ***FLOPs Reduction:*** Consistent channel numbers throughout the network mean that the increase in FLOPs typically associated with deeper layers is mitigated. Since the computational cost for each layer is a function of the input size, filter size, and the number of filters, keeping the number of filters constant across layers ensures that the FLOPs count does not balloon as the network deepens.

By implementing channel unification, Half-UNet becomes a more efficient architecture than U-Net, particularly in terms of computational resources. It requires fewer parameters, making the network leaner and potentially more generalizable, and it demands fewer FLOPs, ensuring that it can run more quickly and cost-effectively, especially on large datasets or when deployed in resource-constrained environments.

Full-Scale Feature Fusion: In U-Net, feature fusion is accomplished through concatenation operations at each level of the upscaling or decoding path. This concatenation takes the high-resolution feature maps from the downsampling path and combines them with the upsampled feature maps from the deeper layers. While this method ensures that the network utilizes both high-level, abstract features and low-level, detailed information, it has a cost.

Half-UNet proposes a full-scale feature fusion approach as an alternative to U-Net's concatenation. This method involves upsampling all feature maps to the size of the original input and then combining them using an additive operation, similar to the residual connections in ResNet. Here's how this change impacts the architecture:

- **Parameter Reduction**: Because the additive operation does not increase the depth of the feature maps (unlike concatenation), the filters in subsequent layers do not need to be as large. Therefore, the network requires fewer parameters. This approach aligns with the principle of parameter efficiency, where the network learns to make the most of the existing feature maps without the need for expanding them.
- **FLOPs Reduction**: The additive operation is computationally inexpensive compared to concatenation followed by convolution. Since the depth of the feature maps does not increase, the FLOPs required for subsequent convolutions are significantly reduced. The addition operation involves a simple element-wise addition, which is far less computationally intensive than the multiplicative operations of convolution.

By implementing full-scale feature fusion, Half-UNet significantly reduces both the number of parameters and FLOPs compared to the standard U-Net. This approach not only simplifies the architecture but also enhances the efficiency of the network, making it better suited for environments where computational resources are limited, without compromising the network's ability to perform accurate segmentation.

Ghost Module Integration:

U-Net: Traditional convolution layers are used extensively throughout U-Net. For each convolution layer, the required parameters and FLOPs can be calculated using the following formulas:

- Parameters (Params) = $(K^2 \times C_{in}+1) \times C_{out}$
- FLOPs = $2 \times K^2 \times C_{in} \times C_{out} \times H_{out} \times W_{out}$

Where K is the kernel size, $C_{in}$and C_{out} are the number of input and output channels respectively, and H_{out} and W_{out} are the height and width of the output feature map.

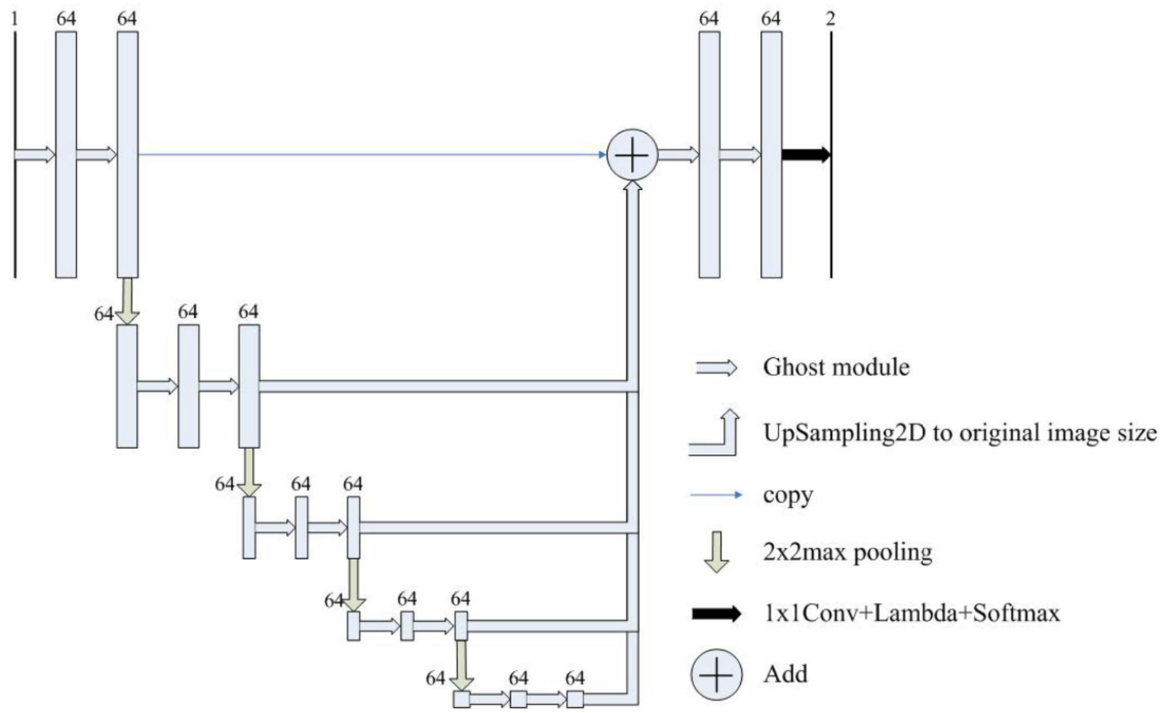
Half-UNet: The Ghost module is integrated to reduce the computational cost. In a Ghost module, only a fraction of the output channels is generated through standard convolution, while the rest are generated through cheaper operations like depthwise separable convolutions. The formulas are altered as follows:

- Parameters (Params) = $[K^2 \times (C_{in}+1)+2] \times C_{out}/2$
- FLOPs = $2 \times K^2 \times (C_{in}+1) \times 2C_{out} \times H_{out} \times W_{out}$

Impact on Parameters and FLOPs:

- **Parameter Reduction**: In Half-UNet, since only half of the feature maps are generated using standard convolutions, and the rest through cheaper operations, the number of parameters is significantly reduced. This is evident from the modified parameter formula, where the factor of $C_{out}/2$ effectively halves the number of parameters needed compared to a traditional convolution layer.
- **FLOPs Reduction**: Similarly, the computational cost is reduced as the FLOPs are nearly halved due to fewer standard convolutions being employed. The factor of $C_{out}/2$ in the FLOPs formula indicates that only half of the output channels contribute to the computational burden typically associated with a full convolutional operation.

The integration of the Ghost module in Half-UNet represents a significant architectural change from U-Net, allowing for a substantial reduction in both parameters and computational load. This makes Half-UNet more efficient and lighter, enabling it to perform complex segmentation tasks with reduced resource requirements.



4 Training

In this section, we delve into the training process of our Half U-Net-based segmentation model tailored to the specific dataset. We discuss data preparation, model architecture, loss function, training configuration, and data augmentation techniques

4.1 Data Preparation

Our dataset comprises input images and corresponding segmentation masks, categorized into training, validation, and testing subsets. To ensure uniformity and ease of processing, we resized all images to a standard resolution of 256x256 pixels. Pixel values were normalized to fall within the [0, 1] range. The masks were binarized to create binary segmentation maps and were also normalized accordingly.

To efficiently handle data loading and preprocessing, we implemented a custom data generator named DataGen. This generator is responsible for reading and processing images and masks, including resizing and necessary preprocessing steps.

4.2 Data Augmentation

In our data augmentation process for the Half UNet model, we primarily focused on introducing rotation invariance and robustness against orientation changes in microscopical images. This is essential when training datasets are limited, to ensure the network learns to recognize and segment features regardless of their orientation. The apply_rotation function is pivotal in this process. It rotates the images by specific angles - 45, 90, 135, 180, 225, 270, and 315 degrees - ensuring comprehensive coverage of possible orientations. The rotation is centered around the image's center, maintaining the original dimensions.

Additionally, we incorporated horizontal and vertical flip transformations through the apply_flip function, which further diversifies the training dataset by simulating mirror images. This function is applied to both the images and their corresponding masks to maintain consistency in training data.

Each image in our dataset undergoes these transformations, creating multiple variants from a single sample. This approach significantly enhances the volume and variety of our training data, contributing to a more robust and generalizable model. The newly generated images, along with their corresponding masks, are saved, thus enriching the dataset with augmented samples that help in improving the model's performance, especially in scenarios with limited annotated samples.

4.3 Loss Function

For training our Half U-Net model, we employed a loss function that combines pixel-wise softmax and cross-entropy loss. This loss function considers the probability of each pixel belonging to a specific class:

$$p_k(x) = \frac{e^{a_k(x)}}{\sum_{k'=1}^{K} e^{a_{k'}(x)}}$$

Where ak(x) represents the activation in feature channel k at pixel position x, and K is the number of classes. The softmax approximation ensures that pk(x) approaches 1 for the class with the maximum activation ak(x) and approaches 0 for all other classes.

To penalize deviations from true labels at each pixel position, we utilized the cross-entropy loss:

$$E = -\sum_x w(x) \log(p'(x)(x))$$

Additionally, we introduced a weighted loss function to assign varying importance to pixels based on their positions within the segmentation masks. This weighted approach encourages the model to prioritize specific regions, such as border pixels between objects of interest.

5 Experiments

In this study, we explored several architectures based on the U-Net model, specifically focusing on the Half-UNet and its variations. The base Half-UNet architecture utilized ghost modules, designed to enhance feature extraction efficiency while reducing computational overhead. Variants of this model included the Half-UNet with L2 regularization to address overfitting concerns, and the Half-UNet with Batch Normalization, implemented to stabilize and expedite the training process. Additionally, we compared these models against the traditional U-Net and a more complex Nested Half-UNet, which was inspired by the U^2-Net architecture, featuring a nested structure for more intricate feature extraction.

5.1 Evaluation Indicators

The IoU metric, also known as the Jaccard Index, is calculated by dividing the intersection of the predicted segmentation and the ground truth by their union. A higher IoU value signifies greater overlap between the model's predictions and the actual data, indicating better segmentation performance.

The F1 score, a harmonic mean of Precision and Recall, is used to assess the model's accuracy. It is computed from the intersection and union of the predicted and true values, where a higher F1 score indicates better accuracy and balance between Precision and Recall.

Recall, or Sensitivity, measures the proportion of actual positives correctly identified by the model. It is especially important in scenarios where missing a positive (such as a medical condition) is critical. Precision quantifies the accuracy of the positive predictions made by the model, highlighting its ability to minimize false positives.

The Dice coefficient, similar in essence to the F1 score, evaluates the model's segmentation accuracy by calculating the ratio of twice the intersection of the predicted and true values to the sum of their sizes. A higher Dice coefficient indicates more accurate segmentation.

5.2 Experimental Results

We compare the proposed Half-UNet with U-Net and variants of U-Net in the task of image segmentation. As shown in the below table,

Model	Accuracy	IoU	F1 Score	Recall	Precision	Dice Coefficient

Basic Half-UNet	**0.9935**	**0.6439**	**0.7501**	**0.8023**	**0.8126**	**0.7743**
U-Net	**0.9926**	**0.6099**	**0.7352**	**0.7568**	**0.8891**	**0.784**
Nested Half-UNet	**0.9942**	**0.6891**	**0.7801**	**0.8236**	**0.8391**	**0.8233**
Half-UNet with L2 Regularization	**0.9924**	**0.5018**	**0.6525**	**0.7243**	**0.7638**	**0.6823**
Half-UNet with Batch Normalization	**0.9943**	**0.6012**	**0.7523**	**0.7991**	**0.8359**	**0.7955**

the Basic Half-UNet model demonstrates commendable accuracy with a value of 0.9935, alongside a respectable Dice Coefficient of 0.7743. Its Intersection over Union (IoU) and F1 Score are recorded at 0.6439 and 0.7501, respectively, illustrating a decent segmentation overlap with the ground truth.

In contrast, the standard U-Net model exhibits a slightly diminished accuracy of 0.9926, yet it surpasses the Basic Half-UNet with a higher Dice Coefficient of 0.784, indicating more precise segmentation capabilities. The Nested Half-UNet, enhancing the architecture further, achieves an accuracy of 0.9942 and the best IoU score among the models at 0.6891, signifying superior segmentation performance, particularly in complex image regions.

The Half-UNet with L2 Regularization shows a mixed performance, with an accuracy of 0.9924 and a lower Dice Coefficient of 0.6823 compared to its counterparts, possibly indicating the effects of over-regularization. Meanwhile, the Half-UNet with Batch Normalization tops the accuracy chart with a value of 0.9943 and boasts an impressive Dice Coefficient of 0.7955, underscoring the effectiveness of batch normalization in enhancing model generalization and segmentation precision.

In summation, these results highlight the nuanced effectiveness of each U-Net variant in image segmentation tasks, with the Half-UNet models (both with and without Batch Normalization) presenting a balanced trade-off between segmentation accuracy and computational efficiency. The reduction in parameters and FLOPs by 98.6% and 81.8%, respectively, when utilizing Half-UNet architectures, points to their potential as cost-effective alternatives in medical image analysis without significantly compromising on performance.

6 Discussion

7 Conclusion

Data AVAILABILITY STATEMENT

Publicly available datasets were analysed in this study.
These datasets can be found at: http://www.eng.usf.edu/cvprg/Mammography/Database.html; https://wiki.cancerimagingarchive.net/display/Public/LIDC-IDRI; http://sourceforge.net/projects/cardiac-mr/.

Reference

Half-UNet:
[Frontiers | Half-UNet: A Simplified U-Net Architecture for Medical Image Segmentation (frontiersin.org)]()
Arxiv:
U2-Net: Going Deeper with Nested U-Structure for Salient Object Detection
Arxiv:
U-Net: Convolutional Networks for Biomedical Image Segmentation