SINERGI Vol. 29, No. 2, June 2025: 447-458
http://publikasi.mercubuana.ac.id/index.php/sinergi
http://doi.org/10.22441/sinergi.2025.2.015

Real-time dental caries segmentation with an efficient Deformable U-Net (DU-Net) for teledentistry system

Zendi Iklima[1]*, Trie Maya Kadarina[1], Ketty Siti Salamah[1], Arrival Dwi Sentosa[2]
[1]Electrical Engineering Department, Faculty of Engineering, Universitas Mercu Buana, Indonesia
[2]School of Electrical Engineering and Informatics, Institut Teknologi Bandung, Indonesia

Abstract
Digital technology has greatly improved teledentistry by facilitating telediagnostics and teleconsultations, particularly benefiting those in remote areas. Additionally, AI advancements enhance diagnostic accuracy and streamline clinical decision-making, reducing costs and resource disparities in dental care. This study presents an improved U-Net architecture, Deformable U-Net (DU-Net), for semantic dental caries segmentation, leveraging deformable convolutions to dynamically adjust sampling points for improved feature extraction and reduced computational redundancy. By connecting encoder-decoder blocks via skip-connections, the DU-Net architecture enables efficient real-time segmentation and balance accuracy while reducing computational demands. The deformable block in DU-Net and DDR U-Net shows a balanced performance and efficiency while maintaining accuracy despite reduced FLOPs. The proposed architecture was implemented in real-time dental caries segmentation on a Dual Core Cortex A72 system and web server. It shows a significant improvement in Dice score, reducing CPU and memory usage compared to conventional U-Net models. Moreover, the DU-Net and its half variants achieved competitive performance with much lower computational demands makes suitable for web servers and embedded applications. The result highlights the DU-Net capability to optimize both computational efficiency and segmentation accuracy, offering a promising solution for real-world applications where speed and resource management are critical, particularly in the medical imaging field.

This is an open access article under the CC BY-SA license

Keywords:
Deformable U-Net;
Dental Caries Segmentation;
Dual Core Cortex A72;
Real-time Segmentation;
Teledentistry;

Article History:
Received: September 28, 2024
Revised: December 17, 2024
Accepted: January 20, 2025
Published: May 15, 2025

Corresponding Author:
Zendi Iklima
Electrical Engineering Department, Faculty of Engineering, Universitas Mercu Buana, Indonesia
Email:
zendi.iklima@mercubuana.ac.id

INTRODUCTION

The advancement of digital technology and communication has a huge influence on enhancing telemedicine services [1, 2, 3], such as teledentistry. Teledentistry is a subset of telemedicine that offers benefits such as ease of dental health diagnostics, teleconsultation, and medical action plan services via digital transformation. Teledentistry provides an alternative for dental health awareness in citizens in remote areas or far from medical facilities [4], thus incurring transportation costs, waiting times, unproductive time, etc. Additionally, dentists can classify the stage of dental caries earlier based on patient dental caries data from smartphone cameras or devices integrated with intraoral cameras. This can show the stages of damage to the dental with high sensitivity; for example, it is specifically designed for the specificity of classifying dental caries at all stages with always a higher than 83.3% [5]. Emphasize teledentity's potential to lower the expenses associated with conventional clinical treatments. Using remote consultations, digital communication, and devices

such as teleconsultation and telemonitoring, practitioners may deliver efficient and cost-effective dental treatment. Integrate teledentistry systems so that dental practitioners can communicate diagnostic and treatment information seamlessly. This fosters collaborative decision-making provides immediate information and aids in treating dental concerns [6].

Deep learning has enabled substantial technical improvements in computer-aided clinic applications, including image classification, automated image segmentation, feature extraction, and image reconstruction. These technologies provide trustworthy diagnostic recommendations by maximizing AI-related technology, assisting radiologists or physicians in clinical decision-making, and reducing their burden. Significantly, artificial intelligence-based technologies can successfully reduce obstacles in medical institutions without radiologists and solve the uneven distribution of medical resources [7]. Convolutional neural networks (CNN) are deep learning architectures that are extensively used in medical imaging such as dental caries recognition and classification [8, 9, 10]. CNN is widely used in medical imaging classification and segmentation as supporting tools for treatment segmentation, enhancing diagnostic accuracy, treatment efficacy, etc. [11, 12, 13]. The classification and segmentation areas have been implemented using CNN. It may be enhanced by analyzing datasets, improving model generalization, optimizing the architecture, and employing transfer learning models (such as VGG16, VGG19 [14], DCNN [15], ResNet, Inception [16], Fast-RCNN [17], DSC [18], DenseNet, EfficientNet [19] and others) that have demonstrated the greatest performance in the classification of dental caries.

However, conventional CNN classification has disadvantages, such as the inability to detect dental caries regions correctly and precisely [20]. Classification models are commonly mapped as binary labels for caries or non-caries, without providing comprehensive information on regions of carious lesions or caries areas. Segmentation models offer the capability to precisely delineate object boundaries within images. Deep learning incorporates the ground truth of dental caries images, and segmentation techniques enable not only the classification of caries presence but also the precise delineation of affected carious zones within the image. The architecture models for segmenting dental caries ought to be established to improve effectively and the precision of automated dental diagnostic systems effectively. Moreover, classification and segmentation methodologies have to be integrated to provide comprehensive assessments of dental health. It is crucial to highlight improvements in clinical applicability to facilitate more effective patient care and management using advanced diagnostic tools. Collectively, these strategies indicate a promising path for the advancement of automated dental caries diagnostics [21, 22, 23].

U-Net is one of the deep learning architectures that is widely used in image segmentation, especially in the biomedical field [24]. U-Net is commonly used in dental caries segmentation [25, 26, 27], liver segmentation [28][29], cardiac MRI segmentation [30][31], bone segmentation [23], retinal vessel segmentation [32], etc. The U-Net model has proven adept at accurately identifying and delineating visual structures, achieving notable average segmentation accuracy. U-Net is a prominent method for biomedical image segmentation. The unique U-shaped architecture is characterized by skip connections that enable the decoder to integrate high-level semantic feature maps with low-level detailed feature maps from the encoder. The encoder decreases the dimensions of the input matrix while augmenting the quantity of feature mappings. The decoder pathway operates inversely to restore the matrix to its initial dimensions by diminishing the number of feature mappings. Consequently, pixel-by-pixel comparisons of segmentation findings to the ground truth are now feasible. U-Net facilitates the transfer of feature maps from each level of the contracting path to the equivalent level in the expanding path, enabling the classifier to analyze features of diverse sizes and complexities [33].

To implement a flexible teledentistry system, we require an advanced embedded system capable of analyzing dental caries images through a portable device. This necessitates the extraction of the U-Net model, which can be efficiently integrated into embedded devices like the Acorn RISC Machine (ARM). By doing so, we can ensure that dental professionals have access to accurate and timely diagnostics, even in remote settings. To achieve efficient and effective real-time segmentation, the balance between accuracy and speed must be improved [34, 35, 36]. Several segmentation models utilize a lightweight encoder and/or decoder to produce low-resolution inputs to obtain high-resolution semantics and provide a lightweight backbone by adjusting conventional CNNs such as Mobile-UNet [37].

Real-time segmentation has been used in several biomedical studies. For example, ColonSegNet trained 880 colonoscopy images on the Kvasair-SEG dataset using the NVIDIA Quadro RTX 6000 to find polyps in 182 frames per

second. It implements data augmentation, fine-tuning, and parameter or layer adjustments to improve segmentation accuracy in case detection, localization, and segmentation itself [38]. Multi-resolution Feature Fusion (MFF) combines auxiliary multi-resolution and multi-channel feature maps with normalization techniques to stabilize inconsistencies in the feature maps. MFF contains Conv-Block, Residual-Block, MFF, Decoder, and Up-Sampling. We trained MFF on the MICCAI 2017 dataset, which includes the Da Vinci XI Surgical System dataset. In real-time performance, MFF runs over NVIDIA GTX 1080Ti compared to U-Net; MFF performs 0.77 fps percent lower than U-Net but it succeeded in surpassing U-Net in accuracy [39].

These preliminary experiments decreased the latency and memory consumption of segmentation models by enhancing higher-resolution feature maps for improved segmentation accuracy. This strategy, however, yields a more complex architecture characterized by a substantial quantity of parameters and FLOPs. This will affect the model's predicting speed in the segmentation area. This work introduces an innovative method for real-time segmentation that improves a streamlined U-Net architecture known as Deformable U-Net. The objective is to develop the U-Net model for implementation in real-time teledentistry systems. The contributions of this paper are:

- Proposed an efficient modified U-Net architecture integrated into deformable dense networks, utilizing a half U-Net framework.
- Enhanced feature extraction: by the use of deformable Utilizing convolutions in the encoder enables the model to analyze various forms, sizes, and scales, resulting in enhanced picture segmentation accuracy.
- Implement a real-time segmentation using the proposed model and modified U-Net variants.
- Proposed two services in the real-time segmentation process, such as segmentation design with embedded devices such as Quad Core Cortex A72 (ARM v8) SoC @ 1.5GHz and segmentation system design using web services.

METHOD

Dental Caries Dataset

The dataset was collected from our dental clinic using an intraoral camera. The dataset included 200 Dental Caries cases generated using an annotation tool (using makesense.ai). Datasets were divided into training, validation, and test sets. The dataset distribution is 67% training, 23% validation, and 10% evaluation to ensure stable model performance and avoid U-Net model overfitting or underfitting. The training set adjusted weights, whereas the validation set determined ideal weights. Finally, the test set provided a comprehensive performance evaluation. Data pretreatment can improve image segmentation model efficiency and durability performance. This research uses data preparation techniques, such as image augmentation which uses random distribution (50% of the original dataset) of the augmentation which contains resizing, rescaling, cropping, and flipping. Augmentation increases image variety and unpredictability to prevent model overfitting. The research approaches utilized in conjunction with the dental caries dataset and the foundational Ground Truth.

Preprocess

The tf.reduce_mean function is used to calculate the average of values in various critical stages. In model training, tf.reduce_mean is used to calculate the average loss across all image vectors, thereby helping to improve model weights optimally. In preprocessing, this function is used to normalize the image by calculating the vector's average, ensuring the input has a stable distribution of values. Additionally, 'tf.reduce_mean' helps with evaluation by calculating the average difference between model predictions and original labels, and is used in the post-processing stage to produce a more stable segmentation result by calculating the average of multiple predictions. This utility ensures the computational efficiency and accuracy of the DU-Net model in detecting dental caries in real time.

Hardware

Real-time dental caries segmentation involves imaging techniques to identify areas of dental caries. This process can be enhanced by leveraging the capabilities of a Quad Core Cortex A72 (ARM v8) SoC @ 1.5GHz microprocessor. The Quad Core Cortex A72 (ARM v8) is designed for efficient processing in embedded systems and has a high-performance architecture. This can facilitate complex computations for processing dental caries image segmentation, effectively identifying carious. The Quad Core Cortex A72 can facilitate high-resolution image processing, where precision is necessary for correct segmentation. The Quad Core Cortex A72 is also capable of running a real-time operating system that facilitates deterministic processing, especially real-time systems that are essential for providing recommendations for consultation and diagnosis.

Design of Experiment

This study presents two real-time segmentation services. The pre-trained model is deployed into an embedded device, namely Quad Core Cortex A72. The model is saved in either '.h5' or '.tensor' file formats. This real-time segmentation service aims to showcase the deployment of the segmentation model on embedded devices such as the Quad Core Cortex A72 via suitable parameter settings. This service utilizes a static pre-trained model; therefore, if you wish to retrain it with a dataset derived from the segmentation area generated by the model, which is aligned with the input image, we also offer a real-time segmentation service through a web interface, specifically employing a straightforward framework such as Flask. Consequently, people or embedded devices may transmit input pictures as photographs, allowing the pre-trained model to execute the segmentation process efficiently. The real-time segmentation service via the web application allows for the retraining of the pre-trained model since the server possesses ample storage to accommodate the fresh dataset produced by the segmentation process by the pre-trained model. Figure 1 represents the two real-time segmentation services.

U-Net

U-Net is characterized by its U-shaped architecture, comprising an encoder structure and a decoder structure. The encoder processes an input image to extract features at several levels of abstraction, encompassing low-level visual attributes (such as texture, color, and form) and high-level semantic characteristics (such as object connections). The decoder uses the encoder's features to produce a feature map that matches the input images' resolution.

The decoder employs deconvolutional layers to up-sample features and recovers their spatial information [40][41]. Figure 2 represents the U-Net architecture in dental caries segmentation, which contains 5 blocks in the encoder and 5 blocks in the decoder.

The feature map of the U-Net encoder and decoder each denotes as X_{En}^{i} and X_{De}^{i}, where i as an index in down-sampling or up-sampling across the encoder or decoder, can be formulated as in (1) [42]:

$$X_{De}^{i} = \{ {}^{X_{En}^{i}, i=N}_{H([C(D(X_{En}^{k}))_{k=1}^{i-1}, C(X_{En}^{i}), C(U(X_{De}^{k}))_{k=i+1}^{N}]), i=1,\ldots N-1} \quad (1)$$

where $H(.)$ is the feature aggregation mechanism which consists of a batch normalization and the activation function. The function $C(.)$ denotes feature map operation containing up-sampling $U(.)$ and down-sampling $D(.)$ operations. The number parameters in the i^{th} U-Net decoder can be computed as in (2) [42]:

$$P_{U-De}^{i} = D_F \times D_F \times [d(X_{De}^{i+1}) \times d(X_{De}^{i}) + d(X_{De}^{i})^2 + d(X_{En}^{i} + X_{De}^{i}) \times d(X_{De}^{i})] \quad (2)$$

where D_F is the kernel size of the convolution, $d(.)$ denotes as the node depth. A set of inter-encode-decode skip connections transmits the low-level tensor from X_{En}^{i} the mini-encoder X_{En}^{i+1}, where a non-overlapping max pooling operation is executed. In U-Net, the decoder skip-connection uses bilinear interpolation to transport high-level tensors from decoder X_{De}^{i} to decoder X_{De}^{i+1}.

The U-Net decoder has a more detailed feature map operation than the Half U-Net decoder, which displays a symmetric decoder. The encoder block indicates the down-sampling process, including five convolutional blocks with an input dimension of 3x3x3. During the convolution process, this block employs the 'he_normal' kernel initializer function to initialize the layer's weights. This approach aims to expedite convergence and enhance model efficacy, particularly when employing the Rectified Linear Unit (ReLU) activation function [22].

A batch normalization layer is employed to stabilize and expedite the training process by normalizing the output of the preceding layer.

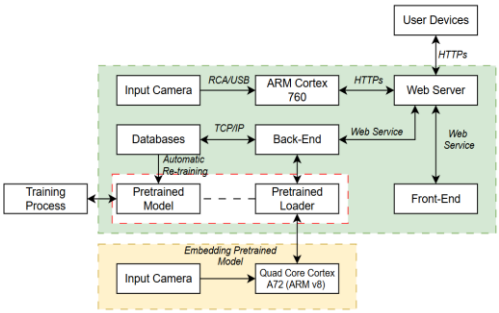

Figure 1. Real-time teledentistry system in Clinic

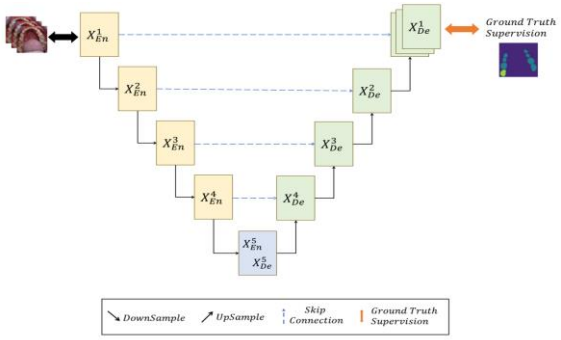

Figure 2. U-Net Architecture

This is accomplished by computing the mean and standard deviation of the data batch during training.

Max pooling is employed to diminish the spatial dimensions of data representation while preserving the most significant aspects. It operates by extracting the greatest value from a specific region of the feature map. Each decoder block employs linear up-sampling to progressively recover the spatial dimensions of the feature map. The spatial dimensions of the feature map are progressively augmented until they match the original input dimensions. Following each up-sampling, a skip connection is established between the current output and the preceding layer with identical spatial dimensions. The up-sampling layer employs linear interpolations that rely on restricted regions. Hence, skip connections can immediately transmit the data acquired during the encoding process to the decoder.

The U-Net and Half-U-Net essentially differ in the number of decoder blocks, with the Half-U-Net incorporating fewer blocks to simplify the model's complexity and decrease its parameter count. This alteration yields a more concise variant of the U-Net design, resulting in a reduced quantity of convolutional layers and parameters in the Half-U-Net relative to its original form. To improve computational efficiency, reduce high losses, and improve the performance of effective models in predicting more accurate segmentation areas [43]. Figure 3 represents the Half U-Net architecture (X_{De}^{3}) in dental caries segmentation.

Deformable U-Net (DU-Net)

Deformable convolution is an advanced technique that enhances the traditional U-Net architecture, particularly in image segmentation tasks. A normal U-Net has an encoder-decoder structure with skip connections and uses standard convolutional layers to pull out features and reduce the size of the input image.

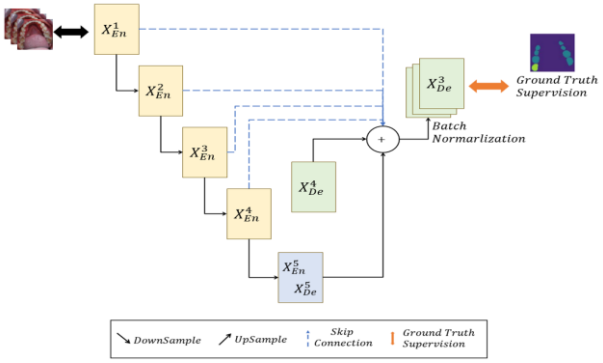

Figure 3. Half U-Net Architecture in (X_{De}^{3})

However, standard convolutions operate on a fixed grid, resulting in static receptive fields that may struggle with complex object shapes and variations in appearance. This limitation can lead to suboptimal performance in precise segmentation tasks. Figure 4 represents the Deformable Dense U-Net architecture in dental caries segmentation.

Combining U-Net architecture with deformable convolution techniques enables the convolutional kernels to dynamically adjust their sampling locations. This flexibility allows the model to focus on the most relevant parts of the input image (ground truth), thereby improving the capture of intricate details and variations in object shape. As a result, deformable convolutions improve the representation of features, which leads to better performance, especially when irregular features or partially obscured objects need to be extracted. Overall, the main difference between deformable and conventional U-Net. Deformable convolutions significantly improve the model's ability to adapt to complex geometries. This adaptability often results in stable training and improved generalization across diverse datasets.

Additionally, applying ReLU activations after each convolution and batch normalization introduces non-linearity and facilitates faster training. The deformable blocks make deformable U-Nets a powerful tool for addressing challenging segmentation scenarios. Deformable Dense U-Net (DU-Net) introduces a new approach to image segmentation with high accuracy and efficient computation, specifically targeting the challenge of image segmentation [40]. Moreover, pattern connectivity in the feature propagation process can enhance the segmentation performance. Figure 5 shows the differentiation between the feature extraction of conventional convolution and the feature extraction of deformable convolution.

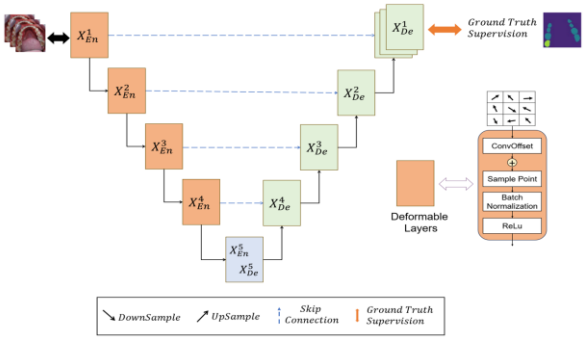

Figure 4. Deformable Dense U-Net Architecture

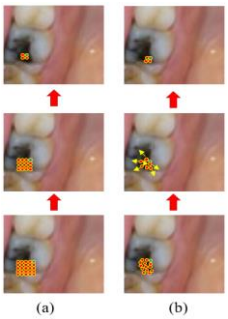

Figure 5. Feature Extraction of (a) Conventional Convolution and (b) Deformable Convolution

The DU-Net architecture can be optimized by adding the residual network. Convolution operations and pooling operations might be affecting the loss of intricate spatial information. To address the information degradation stemming from predefined geometric structures, a residual network is introduced to produce feature maps and to avoid increasing the loss value. This initiative is geared towards enhancing the adaptability of deformable convolution kernels to diverse shapes and positions within the dataset. Bilinear interpolation is employed to compute pixel values at the final sampling location, as offset values often assume fractional values, rendering the deformable convolution sampling locations irregular [22, 41, 44].

Evaluation Metrics

By using geometrically shaped convolution kernels, conventional convolution techniques ignore the shape details of objects in images. In addition, the inclusion of pooling and convolution operations with significant strides results in the loss of complex spatial information. The predefined geometric structure causes information degradation, so we introduce a residual framework to generate feature maps and prevent information loss. This is intended to improve the adaptability of deformable convolution kernels by using bilinear interpolation to calculate the pixel values at the final sampling locations since the offset values often take fractional values in irregular deformable convolution sampling locations.

The Dice coefficient is an important metric for assessing segmentation performance by measuring the overlap between the result of the predicted mask and ground truth. The calculation involves determining the intersection area of the two regions and dividing it by the total size of the two regions. A high Dice score indicates superior alignment/ground truth (as P) and segmentation accuracy (as M), which can be formulated as in (3) [43, 44, 45].

$$Dice = \frac{2 \mid P \cap M \mid}{\mid P \mid + \mid M \mid} \quad (3)$$

RESULTS AND DISCUSSION

Model Performance

This study improves the conventional U-Net performances by adding the deformable and residual network inside the encoder and decoder modules interconnected via skip connections. Deformable U-Net (DU-Net) is designed for effectiveness in feature extraction and segmentation map refinement. This architecture is optimized for faster inference time and reduces the FLOPs and the computational load. So, DU-Net is suitable for real-time applications or environments with limited computational capacity. The main purpose of deformable convolutions is to enhance the ability to extract features efficiently. By dynamically adjusting sampling points, DU-Net minimizes redundancy in feature maps and improves computational efficiency while maintaining high segmentation accuracy. Overall, DU-Net exhibits a strong balance between performance and efficiency and powerful solution for a variety of image segmentation tasks.

Table 1 shows a comparison of the model architectures, representing the number of encoder blocks, the number of decoder blocks, and Floating-Point Operations (FLOPs). Table 1 shows the model architecture comparison, which represents the number of encoder blocks, number of decoder blocks, and Floating-Point Operations (FLOPs).

In terms of computational complexity, the FLOPs metric provides insight into how resource-intensive each model is. The U-Net has the highest FLOPs at $1.69e^{+10}$, but this high complexity does not translate to better performance, as it achieved the lowest Dice score.

Table 1. Model Architectures

Model	Num. En. Block	Num. De. Block	FLOPs
U-Net [25]	5	5	$1.69e^{+10}$
Half U-Net (X_{De}^{4}) [43]	5	2	$6.93e^{+09}$
Half U-Net (X_{De}^{3}) [33]	5	1	$4.95e^{+09}$
DU-Net [40]	5	5	$5.26e^{+09}$
Half DU-Net (X_{De}^{4})	5	2	$5.06e^{+09}$
Half DU-Net (X_{De}^{3})	5	1	$4.61e^{+09}$
DDR U-Net [33]	5	5	$9.56e^{+09}$
DDR Half U-Net (X_{De}^{4}) [33]	5	2	$5.56e^{+09}$
DDR Half U-Net (X_{De}^{3}) [33]	5	1	$3.91e^{+09}$

On the other hand, the Half U-Net (X_{De}^{4}) and Half U-Net (X_{De}^{3}) models have lower FLOPs (6.93e^{+09} and 4.95e^{+10}, respectively) but better performance metrics, which means they use computing resources more efficiently. The DU-Net and its half versions also demonstrate favorable FLOPs, particularly the DU-Net with 5.26e^{+09} and the Half DU-Net (X_{De}^{4}) at 5.06e^{+09}, maintaining competitive performance with significantly lower computational demands. DDR U-Net variant shows the most efficient reduction of the FLOPs (9.56e^{+09} and 3.91e^{+09}, respectively). Practical applications rely on this efficiency, as it enables the deployment of models with fewer resources in environments with limited computational power.

The comparison of FLOPs underscores a key finding: more complex models like the U-Net do not necessarily yield better segmentation results, as seen in their lower Dice scores. In contrast, the Half U-Net and DU-Net models achieve a favorable balance of performance and computational efficiency, making them more viable for real-world applications where speed and resource management are critical. This highlights the importance of architectural optimization in developing effective segmentation models. Figure 6 and Table 2 show the model training logs and training performance.

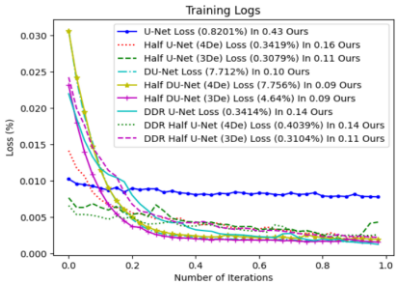

Figure 6. Training Logs

Table 2. Model Training Performances

Model	Dice Coef.	Loss (%)	Exec. Time (hours)
U-Net [25]	0.7963	0.8021	0.43
Half U-Net (X_{De}^{4}) [43]	0.8347	0.3419	0.16
Half U-Net (X_{De}^{3}) [33]	0.8352	0.3079	0.11
DU-Net [40]	0.8639	0.7172	0.10
Half DU-Net (X_{De}^{4})	0.8272	0.7756	0.09
Half DU-Net (X_{De}^{3})	0.8360	0.4640	0.09
DDR U-Net [33]	0.8348	0.3414	0.14
DDR Half U-Net (X_{De}^{4}) [33]	0.8332	0.4039	0.14
DDR Half U-Net (X_{De}^{3}) [33]	0.8491	0.3104	0.11

In terms of segmentation models, the DU-Net stands out with the highest Dice score of 0.8639, indicating superior performance compared to the conventional U-Net, which recorded the lowest Dice score of 0.7963. The Half U-Net variants also demonstrate improved scores, particularly the Half U-Net (X_{De}^{4}) version at 0.8347 and the Half U-Net (X_{De}^{3}) version at 0.8352. The result shows that the DU-Net and its half variant significantly enhances segmentation accuracy.

The segmentation models are evaluated in terms of training losses. The Half U-Net (X_{De}^{3}) achieves the lowest loss at 0.3079% which demonstrates its effectiveness in minimizing error during training. U-Net exhibits a higher training loss of 0.8021%. This highlights an important aspect of model training that lower losses do not necessarily correlate with higher Dice scores and indicates that different models may capture features in different ways. DU-Net and its Half variants have less training time with DU-Net training executed in 0.09 hours, while the full U-Net takes 0.43 hours. According to the FLOPs metric, the Half DU-Net, Half U-Net, and DDR models are capable of maintaining competitive performance with lower computational complexity. This efficiency makes DU-Net and its variants strong candidates for practical applications where speed and accuracy are critical.

Embedded Model Performances

The real-time dental caries segmentation implementation uses a dual-core Cortex A72 (ARM v8) system-on-chip (SoC) operating at 1.5 GHz, paired with 4GB of LPDDR4-2400 SDRAM. The high-performance and energy-efficient design makes the environment ideal for mobile and embedded applications. The system supports H.264 video processing, which allows it to decode at 1080p60 and encode at 1080p30, which is necessary for high-definition, real-time dental caries segmentation. Figure 7 shows the device that was utilized to embed the segmentation models using Quadcore Cortex A72 (ARM v8).

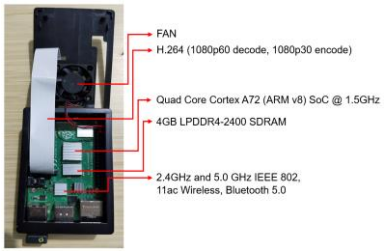

Figure 7. Device with Quadcore Cortex A72 (ARM v8)

A Quadcore Cortex A72 (ARM v8) processor implements the segmentation model with various resolutions (128x128, 256x256, and 512x512 RGB pixels). Figure 8 illustrates the real-time segmentation process on an embedded device, while Table 3 presents the real-time segmentation performance on embedded devices, as measured by the frame-per-second (fps), CPU usage, and memory usage during the stream.

The performance analysis of real-time dental caries segmentation on a dual-core Cortex A72 (ARM v8) architecture reveals notable differences across the models. The U-Net model, with an average frame rate of 30.06 fps, demonstrates significant system resource demands, reflected in a high CPU load of 66.97% and memory usage of 59.70%. While it performs moderately well, the high resource consumption makes it less suitable for resource-constrained environments. In contrast, the Half U-Net models represent efficiency enhancement. The Half U-Net (X_{De}^{4}), reaches an average frame rate of 33.35 fps while using less CPU (57.81%) and memory (48.48%).

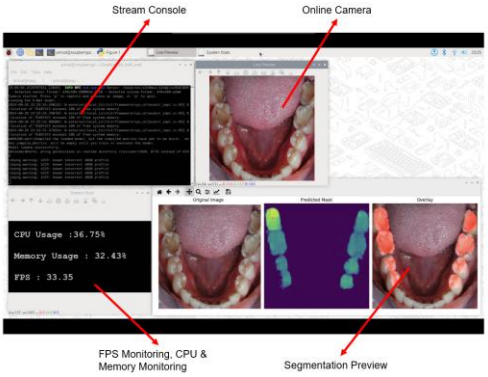

Figure 8. Quadcore Cortex A72 (ARM v8) Segmentation

Table 3. Quad Core Cortex A72 (ARM v8) Segmentation Performances

Model	FPS Avg.	CPU Load Avg. (%)	Mem. Load. Avg. (%)
U-Net [25]	30.06	66.97	59.70
Half U-Net (X_{De}^{4}) [43]	33.35	57.81	48.48
Half U-Net (X_{De}^{3}) [33]	34.01	53.34	42.86
DU-Net [40]	35.59	63.90	46.20
Half DU-Net (X_{De}^{4})	33.29	62.35	45.43
Half DU-Net (X_{De}^{3})	31.02	61.15	44.76
DDR U-Net [33]	26.29	77.00	63.03
DDR Half U-Net (X_{De}^{4}) [33]	28.57	69.71	55.34
DDR Half U-Net (X_{De}^{3}) [33]	35.43	64.40	48.20

The Half U-Net (X_{De}^{3}), performs even better with 34.01 fps, using lower CPU (53.34%) and memory (42.86%) resources. These improvements make Half U-Net models well-suited for real-time applications, particularly in embedded systems where resource management is important.

The DU-Net model achieves the highest frame rate, averaging 35.59 fps, but it still has a relatively moderate CPU load (63.90%) and memory usage (46.20%). DU-Net is highly effective for real-time applications, though it still places a noticeable demand on system resources. The half Du-Net (X_{De}^{4}) runs at 33.29 fps with CPU utilization and memory load each 62.35% and 45.43%. The half Du-Net (X_{De}^{3})runs at 33.29 fps with CPU utilization and memory usage each 62.35% and 45.43%. These models offer a more balanced trade-off between performance and resource efficiency.

The DDR U-Net model delivers the lowest performance metrics which achieves 26.29 fps, the highest CPU load (77.00%), and memory usage (63.03%). This makes DDR U-Net the least viable option for real-time applications, particularly in resource-constrained environments. The DDR Half U-Net (X_{De}^{4}) variant achieves 28.57 fps with a CPU load of 69.71% and memory usage of 55.34%, while the DDR Half U-Net (X_{De}^{3}) variant significantly outperforms with 35.43 fps, a CPU load of 64.40%, and memory usage of 48.20%. Compared to the full DDR U-Net model, the DDR Half U-Net (X_{De}^{3}) offers a better balance between performance and resource consumption.

The Half U-Net model performed the most efficiently, offering strong performance with low resource demands, making it ideal for real-time embedded applications. Although the DU-Net model achieved the highest frame rate, its higher resource usage made it more appropriate for systems with more available processing power. In terms of frame rate and efficiency, the DDR U-Net performed worse, whereas the DDR Half U-Net model offered reasonable performance for systems requiring higher frame rates without exceeding resource usage. These results demonstrate that optimized segmentation architectures can provide real-time processing on embedded platforms with lower resource demands. Dental diagnostics and possibly other medical imaging fields could benefit from their use.

Figure 9, illustrates the implementation of a real-time image segmentation system, leveraging U-Net variants through a web server architecture. The key objective here is to provide live, accurate segmentation of images, which in this case appear

to be dental images from an online camera. The figure captures the integration of different components: the online camera feed, the web server's segmentation process, the browser network console, and the final segmentation preview. This gives a comprehensive view of how images are captured, processed, and segmented in real time, all while maintaining system transparency via network and server logs.

The online camera is the primary source of the images, feeding data into the system for processing. Real-time segmentation demonstrates the ability to support segmentation implementation in dental care or medical diagnostics. The camera captures real-time images of dental lesion damage and then sends 6 captures to a web server. The server then performs a segmentation process on the images. Once the segmentation process is complete, the server sends the segmentation prediction data. Real-time segmentation is performed by the camera continuously sending data to the server for prediction using a trained segmentation model.

The browser network console and the server segmentation process log provide insight into the backend processing and make system performance observable. The browser network console shows the HTTP requests made when images are uploaded to the server. Each image sent to the server is logged in real-time, indicating system performance, network latency, and image upload processing speed. Simultaneously, the server log tracks the segmentation process in real time, showing how each image is segmented and how quickly the segmentation model processes it. The integration of these monitoring tools ensures efficient debugging, and optimization, and ensures that real-time constraints are maintained throughout the system.

Table 4 shows the web server segmentation performances and contains the segmentation process time in seconds per step and the response time average in seconds. We measured the web server segmentation performance by capturing 6 bursts of images with an input size of 512x512x3, each with an average image size of 621 kb and an average FPS of 61 fps. Web server segmentation performance metrics provide information on how well the trained model can perform in real-time applications. Each model's processing time per step and average response time highlight their efficiency and suitability for deployment. For example, Half U-Net (X_{De}^{3}) achieves the fastest segmentation processing time of 1.15 seconds per step, with an average response time of only 3.32 seconds. This efficiency is important in applications where image data processing is time-sensitive such as medical imaging or autonomous systems where delays can have a significant impact on the results.

The DDR U-Net Models produce excellent training performance but have slower segmentation times with an average reaction time of 9.47 seconds. This slower performance can limit its usefulness in real-time applications. Therefore, the longer processing time can cause bottlenecks in scenarios such as rapid and accurate decision-making systems. The variability in response times across architectures highlights the need for not only a high Dice score during training but also for ensuring that the model can efficiently handle incoming data in the context of a web server. Finally, real-time applications are more likely to use models that can deliver both segmentation accuracy and processing speed, ensuring excellent performance while maintaining responsiveness. The correlation between web server segmentation time and training performance highlights the importance of model efficacy in deployment.

In web server situations, models like the DDR U-Net and its variations have long segmentation times that can make real-time processing harder, even though they are very good at training.

Figure 9. Web Server Segmentation

Table 4. Web Server Segmentation Performances

Model	Seg. Proc. Time (s/step)	Avg. Resp. Time (s)
U-Net [25]	2.00	7.22
Half U-Net (X_{De}^{4}) [43]	1.39	3.83
Half U-Net (X_{De}^{3}) [33]	1.15	3.32
DU-Net [40]	1.77	5.40
Half DU-Net (X_{De}^{4})	1.61	4.98
Half DU-Net (X_{De}^{3})	1.55	4.62
DDR U-Net [33]	2.24	9.47
DDR Half U-Net (X_{De}^{4}) [33]	2.06	7.01
DDR Half U-Net (X_{De}^{3}) [33]	1.82	5.62

In contrast, the Half U-Net model exhibits competitive Dice scores during training, resulting in lower average response times and faster processing times. This suggests that the Half U-Net design is designed to use less computational power during training. It can maintain performance while accelerating the speed required for real-time applications. This makes it optimal for situations that require rapid segmentation responses.

CONCLUSION

This study demonstrates significant improvements in efficient, real-time, and accurate semantic segmentation performance. The efficiency of the U-Net decoder block architecture enhances the model's ability to extract features more efficiently, dynamically adjusting sampling points to reduce redundancy. Compared to the conventional U-Net model, the deformable U-Net (DU-Net) is not only faster but also more computationally efficient. These results demonstrate that the architectural trade-off achieved by DU-Net with a simpler computational architecture does not affect segmentation accuracy, as seen in the resulting dice scores. Furthermore, the performance of DU-Net in real-time image segmentation on a Dual Core Cortex A72 device demonstrates the ability of the extracted DU-Net model to be implemented with limited computational resources. The efficiency of the Half DU-Net architecture means that the model can achieve high frame rates while maintaining low CPU and memory usage, making it well-suited for mobile and embedded systems. Real-world applications, such as medical imaging, rely on this efficiency to perform fast and accurate segmentation.

A comparative analysis of different U-Net variants highlights that architectural optimization is key to improving segmentation accuracy and computational efficiency. Segmentation time evaluation in a web server environment confirms that the DU-Net variant consistently outperforms other models in terms of speed and responsiveness. The DU-Net model, characterized by deformable convolutions and efficient architecture, marks an important development in image segmentation. Its real-time performance on embedded systems and web servers illustrates its capabilities for high accuracy and optimization for practical applications, making it a potential option for a variety of real-world segmentation tasks, especially in the field of medical imaging.

ACKNOWLEDGMENT

This research is supported by good cooperation between the Department of Electrical Engineering, Mercu Buana University, and the School of Electrical Engineering and Informatics, Institut Teknologi Bandung. The author would like to express his deepest gratitude to both institutions for the invaluable support of resources, guidance, and encouragement during the research process. Special thanks are also extended to the Ministry of Research, Technology, and Higher Education of the Republic of Indonesia for funding this research. The author's unwavering dedication has played a significant role in achieving the research goals.

REFERENCES

[1] M. Irving et al., “Using teledentistry in clinical practice as an enabler to improve access to clinical care: A qualitative systematic review,” *J Telemed Telecare*, vol. 24, no. 3, pp. 129–146, 2018, doi: 10.1177/1357633X16686776.

[2] M. S. Alauddin, A. S. Baharuddin, and M. I. M. Ghazali, “The Modern and Digital Transformation of Oral Health Care: A Mini Review,” *Healthcare*, vol. 9, no. 2, p. 118, Jan. 2021, doi: 10.3390/healthcare9020118.

[3] Y. Zhang et al., “A Smartphone-Based System for Real-Time Early Childhood Caries Diagnosis,” *Lecture Notes in Computer Science*, vol. 12437 LNCS, pp. 233–242, 2020, doi: 10.1007/978-3-030-60334-2_23.

[4] M. Estai et al., “A systematic review of the research evidence for the benefits of teledentistry,” *J Telemed Telecare*, vol. 24, no. 3, pp. 147–156, 2018, doi: 10.1177/1357633X16689433.

[5] E. K. Kohara et al., “Is it feasible to use smartphone images to perform telediagnosis of different stages of occlusal caries lesions?,” *PLoS One*, vol. 13, no. 9, pp. 1–12, 2018, doi: 10.1371/journal.pone.0202116.

[6] M. R. R. Islam et al., “Teledentistry as an Effective Tool for the Communication Improvement between Dentists and Patients: An Overview,” *Healthcare* vol. 10, no. 8, p. 1586, Aug. 2022, doi: 10.3390/healthcare10081586.

[7] O. Baydar, “The U-Net Approaches to Evaluation of Dental Bite-Wing Radiographs: An Artificial Intelligence Study,” *Diagnostics*, vol. 13, no. 3, 2023, doi: 10.3390/diagnostics13030453.

[8] I. S. Bayrakdar et al., “Deep-learning approach for caries detection and

segmentation on dental bitewing radiographs," *Oral Radiol*, vol. 38, no. 4, pp. 468–479, Oct. 2022, doi: 10.1007/S11282-021-00577-9/metrics.

[9] Y. Liu, K. Xia, Y. Cen, S. Ying, and Z. Zhao, "Artificial intelligence for caries detection: a novel diagnostic tool using deep learning algorithms," *Oral Radiol*, vol. 40, no. 3, pp. 375–384, Jul. 2024, doi: 10.1007/S11282-024-00741-X/METRICS.

[10] H. Zhu, Z. Cao, L. Lian, G. Ye, H. Gao, and J. Wu, "CariesNet: a deep learning approach for segmentation of multi-stage caries lesion from oral panoramic X-ray image," *Neural Comput Appl*, vol. 35, no. 22, pp. 16051–16059, 2023, doi: 10.1007/s00521-021-06684-2.

[11] W. Yao et al., "From CNN to Transformer: A Review of Medical Image Segmentation Models," *J Imag Inform Med*, vol. 37, no. 4, pp. 1529–1547, Mar. 2024, doi: 10.1007/S10278-024-00981-7/METRICS.

[12] J. H. Lee et al., "Detection and diagnosis of dental caries using a deep learning-based convolutional neural network algorithm," *J Dent*, vol. 77, pp. 106–111, Oct. 2018, doi: 10.1016/j.jdent.2018.07.015.

[13] V. Geetha et al., "Dental caries diagnosis in digital radiographs using back-propagation neural network," *Health Inf Sci Syst*, vol. 8, no. 1, 2020, doi: 10.1007/s13755-019-0096-y.

[14] D. Saini, R. Jain, and A. Thakur, "Dental Caries early detection using Convolutional Neural Network for Tele dentistry," *2021 7th Int. Conf. on Adv. Comp. and Comm. Syst, ICACCS 2021*, pp. 958–963, 2021, doi: 10.1109/ICACCS51430.2021.9442001.

[15] A. Sonavane, R. Yadav, and A. Khamparia, "Dental cavity classification of using convolutional neural network," *IOP Conf Ser Mater Sci Eng*, vol. 1022, no. 1, 2021, doi: 10.1088/1757-899X/1022/1/012116.

[16] M. Moran, M. Faria, G. Giraldi, L. Bastos, L. Oliveira, and A. Conci, "Classification of approximal caries in bitewing radiographs using convolutional neural networks," *Sensors*, vol. 21, no. 15, pp. 1–12, 2021, doi: 10.3390/s21155192.

[17] A. Laishram and K. Thongam, "Detection and classification of dental pathologies using faster-RCNN in orthopantomogram radiography image," *2020 7th International Conference on Signal Processing and Integrated Networks, SPIN 2020*, pp. 423–428, 2020, doi: 10.1109/SPIN48934.2020.9071242.

[18] T. M. Kadarina, Z. Iklima, R. Priambodo, Riandini, and R. N. Wardhani, "Dental caries classification using depthwise separable convolutional neural network for teledentistry system," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 2, pp. 940–949, 2023, doi: 10.11591/eei.v12i2.4428.

[19] F. Oztekin et al., "An Explainable Deep Learning Model to Prediction Dental Caries Using Panoramic Radiograph Images," *Diagnostics*, vol. 13, no. 2, 2023, doi: 10.3390/diagnostics13020226.

[20] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," *Lecture Notes in Computer Science*, vol. 9351, pp. 234–241, 2015, doi: 10.1007/978-3-319-24574-4_28.

[21] V. Majanga and S. Viriri, "A Survey of Dental Caries Segmentation and Detection Techniques," *Scientific World Journal*, vol. 2022, 2022, doi: 10.1155/2022/8415705.

[22] Z. Iklima et al., "Dental Caries Segmentation using Deformable Dense Residual Half U-Net for Teledentistry System," *Journal of Electronics, Electromedical Engineering, and Medical Informatics*, vol. 6, no. 4, pp. 489–498, Sep. 2024, doi: 10.35882/jeeemi.v6i4.511.

[23] M. Fradi, E. Zahzah, and M. Machhout, "Real-time application based CNN architecture for automatic USCT bone image segmentation," *Biomed Signal Process Control*, vol. 71, Jan. 2022, doi: 10.1016/j.bspc.2021.103123.

[24] N. S. Punn and S. Agarwal, "Modality specific U-Net variants for biomedical image segmentation: a survey," *Artif Intell Rev*, vol. 55, no. 7, pp. 5845–5889, Oct. 2022, doi: 10.1007/s10462-022-10152-1.

[25] A. Fariza, A. Z. Arifin, and E. R. Astuti, "Automatic Tooth and Background Segmentation in Dental X-ray Using U-Net Convolution Network," *2020 6th Int. Conf. Sci. Inform. Tech. ICSITech 2020*, September 2021, pp. 144–149, 2020, doi: 10.1109/ICSITech49800.2020.9392039.

[26] I. S. Bayrakdar et al., "A U-Net Approach to Apical Lesion Segmentation on Panoramic Radiographs," *Biomed Res Int*, vol. 2022, 2022, doi: 10.1155/2022/7035367.

[27] S. Lee et al., "Deep learning for early dental caries detection in bitewing radiographs," *Sci Rep*, vol. 11, no. 1, pp. 1–8, 2021, doi: 10.1038/s41598-021-96368-7.

[28] R. A. Khan, Y. Luo, and F. X. Wu, “RMS-UNet: Residual multi-scale UNet for liver and lesion segmentation,” *Artif Intell Med, vol. 124, Feb. 2022,* doi: 10.1016/j.artmed.2021.102231.

[29] X. Li et al., “H-DenseUNet: Hybrid Densely Connected UNet for Liver and Tumor Segmentation from CT Volumes,” *IEEE Trans Med Imaging*, vol. 37, no. 12, pp. 2663–2674, 2018, doi: 10.1109/TMI.2018.2845918.

[30] T. Wang et al., “ICA-UNet: ICA Inspired Statistical UNet for Real-Time 3D Cardiac Cine MRI Segmentation,” *Lecture Notes in Computer Science*, vol. 12266 LNCS, pp. 447–457, 2020, doi: 10.1007/978-3-030-59725-2_43.

[31] T. Wang et al., “MSU-Net: Multiscale Statistical U-Net for Real-Time 3D Cardiac MRI Video Segmentation,” *Lecture Notes in Computer*, vol. 11765 LNCS, pp. 614–622, 2019, doi: 10.1007/978-3-030-32245-8_68.

[32] D. Huang, H. Guo, and Y. Zhang, “ADD-Net:Attention U-Net with Dilated Skip Connection and Dense Connected Decoder for Retinal Vessel Segmentation,” *Lecture Notes in Computer Science*, vol. 13002 LNCS, pp. 327–338, 2021, doi: 10.1007/978-3-030-89029-2_26.

[33] T. M. Kadarina et al., “A simplified dental caries segmentation using Half U-Net for a teledentistry system,” *SINERGI*, vol. 28, no. 2, pp. 251–258, Apr. 2024, doi: 10.22441/sinergi.2024.2.005.

[34] Y. Nirkin, T. Hassner, and F. Ai, “HyperSeg: Patch-wise Hypernetwork for Real-time Semantic Segmentation,” Oct. 2021, doi: 10.1109/CVPR46437.2021.00405.

[35] G. Li, I. Yun, J. Kim, and J. Kim, “DABNet: Depth-wise Asymmetric Bottleneck for Real-time Semantic Segmentation,” *Int J Multimed Inf Retr*, no. 13, Jul. 2024, doi: 10.1007/s13735-024-00321-z.

[36] R. Robinson et al., “Real-Time Prediction of Segmentation Quality,” in *Lecture Notes in Computer Science*, Springer Verlag, 2018, pp. 578–585. doi: 10.1007/978-3-030-00937-3_66.

[37] H. S. Yoon, S. W. Park, and J. H. Yoo, “Real-time hair segmentation using mobile-unet,” *Electronics (Switzerland)*, vol. 10, no. 2, pp. 1–12, Jan. 2021, doi: 10.3390/electronics10020099.

[38] D. Jha et al., “Real-Time Polyp Detection, Localization and Segmentation in Colonoscopy Using Deep Learning,” *IEEE Access*, vol. 9, pp. 40496–40510, 2021, doi: 10.1109/access.2021.3063716.

[39] M. Islam et al., “Real-time instrument segmentation in robotic surgery using auxiliary supervised deep adversarial learning,” *IEEE Robot Autom Lett*, vol. 4, no. 2, pp. 2188–2195, Apr. 2019, doi: 10.1109/LRA.2019.2900854.

[40] Q. Jin et al., “DUNet: A deformable network for retinal vessel segmentation,” *Knowl Based Syst*, vol. 178, pp. 149–162, 2019, doi: 10.1016/j.knosys.2019.04.025.

[41] F. Li, X. Liu, Y. Yin, and Z. Li, “DDR-Unet: A High-Accuracy and Efficient Ore Image Segmentation Method,” *IEEE Trans Instrum Meas*, vol. 72, 2023, doi: 10.1109/TIM.2023.3317480.

[42] H. Huang et al., “UNet 3+: A Full-Scale Connected UNet for Medical Image Segmentation,” *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2020, no. iii, pp. 1055–1059, doi: 10.1109/ICASSP40776.2020.9053405.

[43] H. Lu, Y. She, J. Tie, and S. Xu, “Half-UNet: A Simplified U-Net Architecture for Medical Image Segmentation,” *Front Neuroinform*, vol. 16, no. June, pp. 1–10, 2022, doi: 10.3389/fninf.2022.911679.

[44] S. Saumiya and S. W. Franklin, “Residual Deformable Split Channel and Spatial U-Net for Automated Liver and Liver Tumour Segmentation,” *J Digit Imaging*, vol. 36, no. 5, pp. 2164–2178, 2023, doi: 10.1007/s10278-023-00874-1.

[45] H. Zhou, H. Leung, and B. Balaji, “AR-UNet: A Deformable Image Registration Network with Cyclic Training,” *IEEE/ACM Trans Comput Biol Bioinform*, 2023, pp. 1–10, doi: 10.1109/TCBB.2023.3284215.