AI-Enabled Vessels Segmentation Model for Real-Time Laparoscopic Ultrasound Imaging

Ignas Kupcikevicius*[1, 4], Luca Boretto[1], Inger Annett Grünbeck[1, 4], Rahul Prasanna Kumar[1], Varatharajan Nainamalai[1], Seyed Mohammadmehdi Sadat Akhavi[1, 3], Bjørn Edwin[1, 2, 3], and Ole Jakob Elle[1, 4]

[1]The Intervention Center, Rikshospitalet, Oslo University Hospital, Oslo, Norway
[2]Department of Hepato-Pancreatic-Biliary Surgery, Oslo University Hospital, Oslo, Norway
[3]Institute of Clinical Medicine, Faculty of Medicine, University of Oslo, Oslo, Norway
[4]Department of Informatics, University of Oslo, Oslo, Norway

Abstract

Laparoscopic ultrasound (LUS) is essential for assessing the liver during laparoscopic liver resections. However, the interpretation of LUS images presents significant challenges due to the steep learning curve and image noise. In this study, we propose an enhanced U-Net-based neural network with a ResNet18 backbone specifically designed for real-time liver vessel segmentation of 2D LUS images. Our approach incorporates five preprocessing steps aimed at maximizing the training information extracted from the ultrasound sonogram region. The modified U-Net model achieved a Dice coefficient of 0.879, demonstrating real-time performance at 40 frames per second and enabling the development of advanced ultrasound-based surgical navigation solutions.

1 Introduction

Liver cancer remains one of the top 10 deadliest cancers worldwide, resulting in approximately 750,000 annual deaths [1]. The reason for its mortality rate is late diagnoses, limited treatment options, and underlying liver disease with aggressive tumor biology [2]. To locate liver tumors and vessels during laparoscopic liver surgery, clinicians are using the laparoscopic ultrasound (LUS), which helps to navigate and to avoid unnecessary damage during liver resection or ablation. LUS is a radiation-free medical device, portable and cost-effective. It provides real-time images by capturing ultrasound reflected pulses from soft tissues and bones [3]. All of these LUS benefits give clinicians the ability to effectively diagnose liver cancer, such as hepatocellular carcinoma and other metastases. Additionally, LUS allows visualization of essential liver structures, the portal vein, hepatic veins, and bile ducts.

While valuable, LUS comes with several drawbacks. A key problem is speckle noise, an artifact from ultrasound waves, that interferes as reflected off tissue microstructures. This effect lowers overall image quality [4]. Vessel boundaries can also appear unclear because of differences in tissue echogenicity - the way tissues reflect sound. When boundaries fade, tracking blood vessels during surgery becomes more difficult. Finally, underlying conditions like fatty liver or cirrhosis are causing liver texture changes which interfere with the interpretation of ultrasound scans [5].

These imaging issues limit how effectively LUS can guide surgeons during liver procedures [6]. Several conventional techniques could be used to account for these challenges. One of the default modes of current ultrasound (US) systems, is Color Doppler mode, which can be used to visualize blood flow by detecting frequency shifts in moving blood cells and to enable real-time assessment of vascularity. However, it has a relatively small region of interest, and its effectiveness is heavily dependent on the operators' skill, which might introduce inconsistency in the interpretation of the LUS data [7].

Another traditional visualization method is a Contrast-Enhanced Ultrasound (CEUS). It uses microbubble contrast agents to improve the visibility of blood vessels. Unfortunately, this method requires careful timing to capture the best blood flow enhancement after the contrast is given, which can be difficult in busy surgical environment [8]. Traditional segmentation algorithms, such as region growing, thresholding, and clustering, have also been employed for tissue segmentation [9]. All of them require manual tuning of thresholds value and seed points, which limits their robustness in handling the complex and heterogeneous tissue structures present in ultrasound images.

1.1 Related work

Over recent years, deep learning has become a common approach for automated vessel segmentation. Reported performance from different studies varies widely across imaging modalities, with Dice scores of 0.734 for ultrasound [10], 0.928 for MRI [11], and 0.814 for CT [12]. Dice scores come from different datasets and are not directly comparable, however, they show that ultrasound remains a challenging

*Corresponding Author.

Proceedings of the 7th Northern Lights Deep Learning Conference (NLDL), PMLR 307, 2026.

modality for vessel segmentation. U-Net-based architectures, have been recognized as the gold standard for semantic segmentation tasks [13]. Their encoder-decoder structure and skip connections have made them adaptable to enhancements such as adding residual blocks (ResU-Net) [14], dense connections (DenseU-Net) [6], attention gates (Attention U-Net) [15], or transformers (TransU-Net) [16]. Although these studies have demonstrated competitive results in segmenting various biological tissues from ultrasound data, only a few have explored the performance of real-time segmentation [6, 17].

Real-time ultrasound image segmentation is a complex task due to the noise and inconsistent data. Preprocessing is often employed to suppress speckle noise and reduce artifacts, but this adds computational overhead. Varying echogenicity makes boundary detection difficult. This means the model requires careful tuning to remain reliable under these imaging conditions. Post-processing techniques, such as mask refinement for frame-to-frame consistency, or resizing output to the original resolution, add further computational load. This makes it difficult to balance between models tuned for accuracy and models tuned for the speed required in real-time applications. Smistad et al. [18] used an Artificial Intelligence (AI) model to segment blood vessels, nerves, and bone structures during anesthesia-related procedures, and showed a promising real-time performance. However, the predictions were made on a frame-by-frame basis without considering the temporal information in the sequential ultrasound data, which could have inherent potential information to enhance the performance.

1.2 Contribution summary

This paper presents an automated AI-enabled LUS model for real-time vessel segmentation, developed to support liver surgery and improve intraoperative guidance. Our main contribution is an end-to-end, real-time workflow that automatically extracts and masks the ultrasound sonogram, applies CLAHE tuned to our LUS data, and uses a triplet-frame input with a lightweight ResNet18 backbone. Together, these components improve vessel detection and segmentation continuity while keeping inference speed compatible with intraoperative use. Below, we outline the methodological and dataset contributions that form this clinically-oriented pipeline.

1. Fully anonymized LUS liver video data was locally acquired and annotated with the assistance of experienced clinicians, and all annotations were verified and approved by a radiologist.

2. A dynamic approach was developed to extract the ultrasound sonogram from video frames. It enabled precise masking of the imaging area and prevented the network from learning irrelevant background features, thereby improving segmentation accuracy without compromising real-time performance.

3. The triplet input setup, similar to ones used for LUS-CT co-registration [19] and for object recognition [20], was integrated into a lightweight ResNet18 U-Net model, enhancing segmentation quality by introducing contextual information between frames.

4. Contrast Limited Adaptive Histogram Equalization (CLAHE) [21] was applied and optimized for our dataset. It enhanced vessel boundaries and improved lumen visibility, which resulted in increased segmentation accuracy.

5. A comprehensive study was conducted to evaluate the performance of different U-Net family encoders, focusing on both segmentation accuracy and real-time inference efficiency.

The AI-generated 2D liver vessel segmentation masks can also be used for 3D vessel reconstruction, aiding in image registration between preoperative and intraoperative stages.

2 Proposed methodology

The proposed method uses an encoder–decoder architecture with a ResNet18 backbone to perform real-time vessel segmentation in laparoscopic ultrasound frames.

2.1 Pre-processing pipeline

As shown on the left side of Figure 1, all the LUS frames undergo standardization before entering the network to: a) emphasize learning focused on the acoustic sonogram, b) stabilize contrast across different cases, acquisition depths, and sonogram shapes, and c) ensure a consistent input shape while preserving spatial geometry. The specific preprocessing steps (1 to 5) visualized in Figure 1 will be described in detail as follows:

1. **Grayscale conversion**: Our recorded LUS videos contain identical R, G and B channels (stacked intensity), therefore, we converted frames to a single grayscale channel to remove redundancy, reduce computation and memory use, and avoid learning artificial color patterns that are not part of the actual ultrasound signal.

2. **Sonogram detection:** To locate the US sonogram as a trapezoid in frame coordinates, the dynamic contour-based detection with a fallback heuristic was developed and used. For the fallback, the most recently detected good coordinates

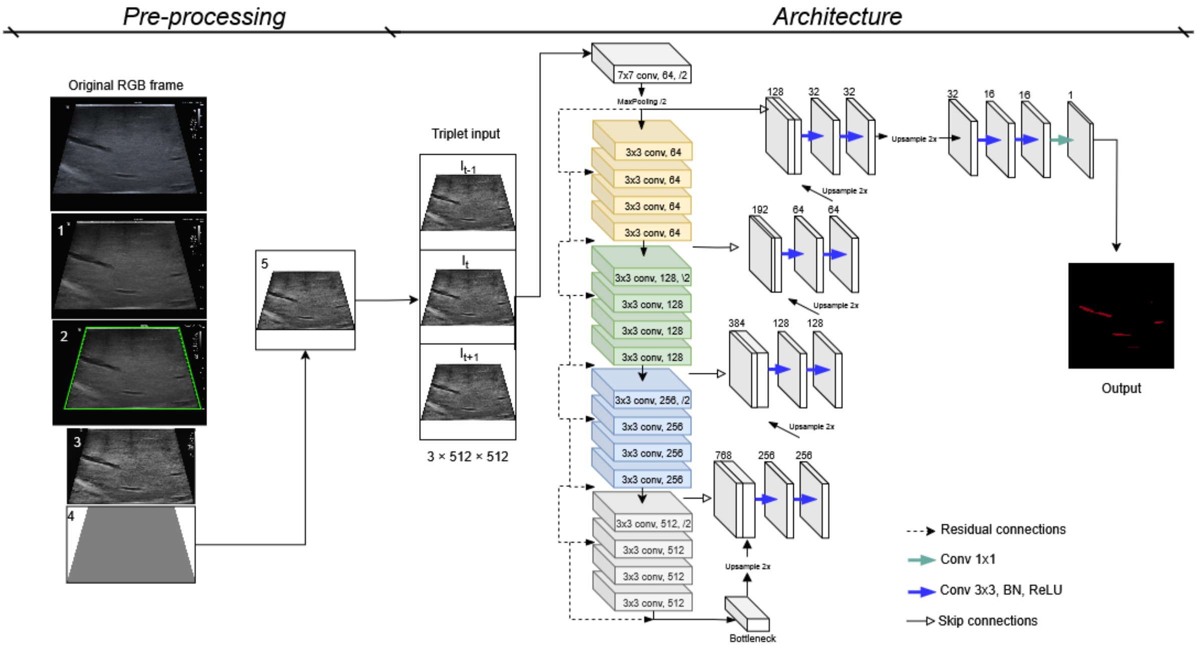

Figure 1. Overview of the proposed vessel segmentation pipeline. The raw RGB laparoscopic ultrasound frames undergo five preprocessing steps, are arranged into temporal triplets, encoded with a ResNet18 backbone, and decoded with a U-Net decoder to generate the final binary mask of the vessels.

were used. The trapezoid is defined by four corner points, stored in a 4×2 matrix $P^{\text{trap}} \in \mathbb{Z}_{\geq 0}^{4\times 2}$, where each row holds the (x, y) coordinates of one corner in a set of intigers $\mathbb{Z}$. By using this locally developed method, we managed to successfully detect the sonogram contour even when the acquisition depth was changing.

3. **Tight crop and CLAHE:** After locating the sonogram coordinates, we applied a tight rectangular crop to remove unnecessary background. Following the CLAHE application strategy of Ansari et al. [6], we tuned two hyperparameters - the contrast limiting threshold (clipLimit) and the grid size for histogram equalization (GridSize), using a small-scale grid search on a validation subset. The selected values, clipLimit = 2.0 and GridSize = 8×8, consistently improved image contrast and produced better segmentation performance both quantitatively and qualitatively.

4. **Sonogram masking:** After cropping, the sonogram was isolated using a binary polygon mask (sonogram = 0, outside = 255). Whitening the background region prevents the model from learning irrelevant background patterns and reduces false positives outside the imaging area, ensuring the network focuses on anatomical features inside the sonogram. The original video frames contained non-anatomical elements such as text overlays, depth and distance markers, and interface graphics from the ultrasound device. Including these elements during training could lead the network to associate them with anatomical structures, so removing them ensured that only clinically relevant image content was used. Similar masking-based extraction approaches are commonly applied in ultrasound preprocessing [22].

5. **Resizing and padding:** In the final step, we resized the ultrasound frame while preserving its aspect ratio to avoid geometric distortion of vessel structures, which is a standard operation in ultrasound imaging workflows [22]. After resizing, the image was symmetrically padded to 512 x 512 using bilinear interpolation to provide a fixed square input compatible with a wide range of network backbones.

By applying these five preprocessing steps to each LUS video frame, we direct the model's focus toward the sonogram area that contains vascular anatomy. CLAHE enhances local contrast for thin, low-contrast vessels, while aspect-ratio-based resizing prevents geometric distortions of tubular structures.

2.2 Network architecture

For this study, we tested and adopted a U-Net-based encoder–decoder network with multiple backbones from the U-Net family, including lightweight ResNet18 [23], MobileNet_v2 [24], DenseNet-121 [25], medium-sized ResNet50 [23], and Vanilla U-Net [13], as well as a larger model, InceptionNetV2 [26]. All encoders except the vanilla U-Net were used via the Segmentation Models PyTorch library [27], which provides standardized open source implementations of these architectures. The selected encoder, ResNet18, and the standard U-Net decoder, used for training and inference, are shown on the right side of Figure 1. ResNet18, pretrained on ImageNet [28], provides residual blocks that support stable training on small medical datasets. To exploit temporal coherence, sequential frames are processed as triplets rather than individually, enabling motion-aware and more consistent predictions. In general, U-Net architecture was selected due to its strong performance in segmentation tasks and its skip connections which help retain fine spatial details that are often lost during the downsampling process.

2.2.1 Modified input layer: temporal triplet setup

Unlike conventional U-Net inputs that use a single 2D frame, we implemented triplets of consecutive frames to utilize the temporal information of sequential ultrasound video data. During training process, we used a symmetric triplet setup $[I_{t-1}, I_t, I_{t+1}]$, with the ground truth mask corresponding to the middle image I_t. This configuration allows the model to learn context from both past and future frames, possibly enhancing vessel continuity and robustness to speckle noise. During inference, since future frames are not available, we switched to a more standard triplet setup: $[I_{t-2}, I_{t-1}, I_t]$, which maintains the benefits of temporal context without compromising real-time performance.

2.2.2 Loss, optimization and training controls

In preliminary experiments with a Vanilla U-Net, we evaluated several segmentation loss functions commonly used in medical imaging. Dice loss [29], Focal loss [30], and Binary Cross Entropy (BCE) loss [31] functions were tested and compared. By looking at the Dice score curves in Figure 2 we noticed, that BCE was more stable and it reached slightly higher Dice values than Dice or Focal loss. Dice and Focal loss showed more fluctuations, suggesting less reliable optimisation. BCE was therefore selected as the primary training loss for all encoder backbones, as it offered the most consistent generalization and improvements. The Dice coefficient was used as the primary evaluation metric during model validation. The BCE loss is defined as follows:

$$\mathrm{BCE}(p, y) = -\left(y \log(p) + (1-y)\log(1-p)\right), \quad (1)$$

where $y \in \{0, 1\}$ is the ground truth, and $p \in [0, 1]$ is the predicted probability (after sigmoid function). We employed Binary Cross-Entropy with logits loss (*BCEWithLogitsLoss* in PyTorch [32]), which is equivalent to applying a sigmoid activation followed by binary cross-entropy, but implemented in a numerically more stable form.

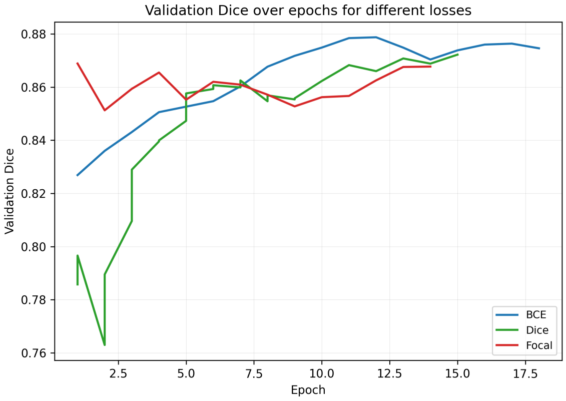

Figure 2. Validation Dice across epochs for three different loss functions (BCE, Dice, and Focal), using early stopping. These curves are smoothed, so the last few points may appear to increase slightly even though the underlying validation Dice already plateaued when early stopping triggered.

2.2.3 Optimization

All our networks were trained for up to 100 epochs under identical conditions to ensure a fair comparison across backbones. For optimization, we used Adam optimizer with a learning rate of 1×10^{-3} and a learning rate scheduler (factor = 0.5, patience = 5) to improve convergence once the validation loss plateaued. During preliminary experiments, we evaluated several geometric and intensity-based augmentation strategies. Different encoder architectures responded inconsistently to these augmentations, with some showing improved performance and others degrading under the similar augmentation setup. To avoid introducing architecture-dependent bias and to maintain a controlled comparison across models, we therefore trained all networks without additional augmentation. To accelerate training and reduce GPU memory usage, automatic mixed precision was used. We used early stopping function to interrupt training after 10 epochs without improvement in the validation Dice score. The batch size was set to 18, determined by the available GPU memory (RTX 4080, 12 GB VRAM). Each channel of the triplet input was normalized to $[-1, 1]$to improve training stability.

2.3 Dataset and data split

The dataset consists of laparoscopic ultrasound videos from 11 acquisitions, obtained from 9 patients. One patient contributed three acquisitions from separate sessions, each capturing distinct liver views, as confirmed by a radiologist, and treated as independent cases. By doing this we preserved

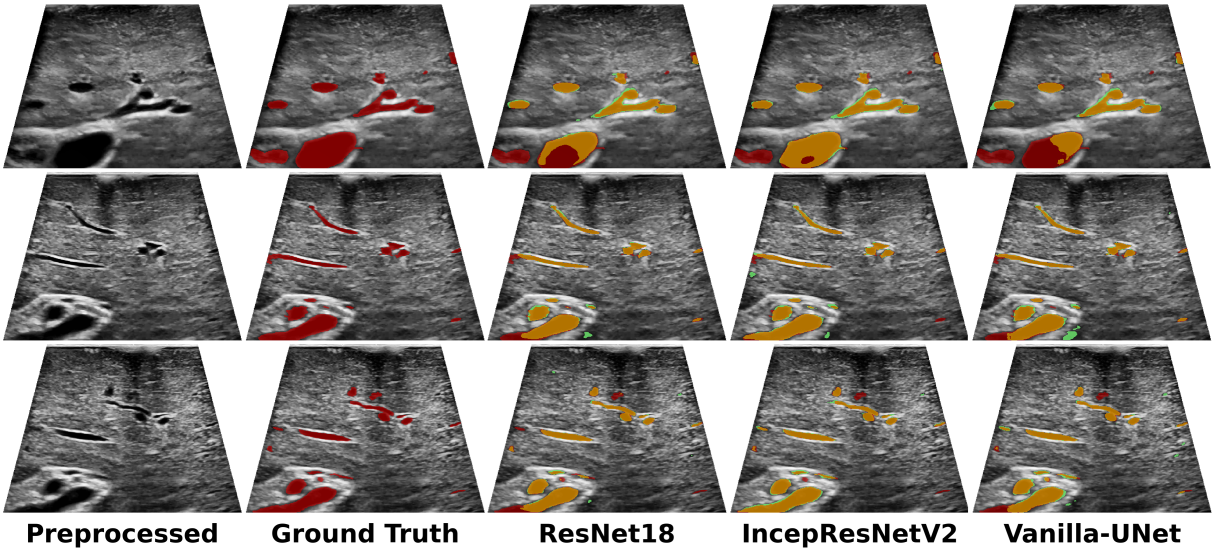

Figure 3. Vessel segmentation results using three models.
Red: ground truth, Yellow: model predictions, Green: over-predicted regions.
.

all usable data and maximized the full size of the training set. All data were recorded using a commercial laparoscopic ultrasound system, and only video-format data were available. Access to raw ultrasound signals was not possible. From these videos, 2,200 sequential frames containing vessels were extracted, and pixel-wise binary masks were created and subsequently verified by a radiologist. To preserve independence between development and evaluation, one case was set aside as the test set. The remaining ten acquisitions were used for model development under a 5-fold cross-validation scheme, with eight used for training and two for validation in each fold. After cross-validation, a final model was trained on the same ten acquisitions using a 90/10 split to maximize the training sample while retaining an internal validation subset. Final performance was evaluated on the held-out test set.

3 Results and discussion

Various backbone architectures were tuned and compared within the proposed U-Net framework. Performance was evaluated using the cross validation protocol described in Section 2.3 to identify the most promising encoders. A final model was then trained and assessed on the held out test set to measure generalization on unseen data.

3.1 Quantitative results

Comparison results presented in Table 1 are from 5-fold cross-validation experiments, made with six backbone configurations. All tested encoders achieved Dice scores above 0.9 in 5-fold CV, demonstrating that vessel segmentation in LUS is a learnable task. The consistent results across architectures indicate robustness and suggest that encoder selection can be made not only by the accuracy alone, but also by efficiency and deployment factors. When looking at the results from the test set, both InceptionResNetV2 and ResNet18 performed well, with InceptionResNetV2 reaching the highest Dice score of 0.888 and recall of 0.867. Deep architecture and Inception modules allow InceptionResNetV2 to capture multi-scale features, while residual connections help stabilize training similarly as in ResNet18. However, with a large number of parameters, the model is computationally heavy, leading to longer training times and slower inference compared to lighter encoders.

Despite being the lightest model, ResNet18 produced competitive Dice of 0.879 and high precision score of 0.923, benefiting from residual connections that support efficient gradient flow and stable feature learning. Additionally, the smaller parameter count noticeably improved the inference speed, preserving real-time capability even with our additional preprocessing steps. This balance of accuracy and efficiency makes ResNet18 particularly suited for laparoscopic ultrasound vessel segmentation, where reliable performance must be achieved under strict computational constraints.

3.2 Qualitative results

Selected segmentation examples from three test frames, shown in Figure 3, compare ground truth annotations with predictions from the lightweight ResNet18, the heavier InceptionResNetV2, and the medium-sized Vanilla U-Net. Across the models, there is a noticeable tendency for slight over-

Table 1. Comparison of vessel segmentation performance of different models: Dice coefficient (DC) with standard deviation (Std) from 5-fold cross-validation (CV), and DC, recall, precision, and Intersection over Union (IoU) on the test set.

Encoder Type	5-Fold CV	Test set			
	DC ± Std	DC	Recall	Precision	IoU
ResNet18	0.916 ± 0.002	0.879	0.840	**0.923**	0.783
ResNet50	0.904 ± 0.002	0.856	0.811	0.912	0.748
MobileNet_v2	0.906 ± 0.002	0.862	0.819	0.917	0.757
Vanilla U-Net	0.918 ± 0.003	0.859	0.827	0.897	0.753
DenseNet121	0.901 ± 0.003	0.849	0.798	0.913	0.738
IncResNetV2	0.926 ± 0.002	**0.888**	**0.867**	0.911	**0.798**

Table 2. Comparison of single-frame and triplet input ResNet18 U-Net models. Mean ± standard deviation is reported for all metrics. Arrows indicate whether higher (↑) or lower (↓) values are better.

Metric	ResNet18 U-Net (Single)	ResNet18 U-Net (Triplet)
Per-frame DC vs. GT (↑)	0.875 ± 0.031	**0.879 ± 0.030**
Temporal DC (↑)	0.932 ± 0.056	**0.938 ± 0.045**
Temporal IoU (↑)	0.877 ± 0.091	**0.886 ± 0.073**
Flip rate (↓)	0.008 ± 0.008	**0.007 ± 0.007**
Boundary jitter (px, ↓)	1.334 ± 1.257	**0.980 ± 0.990**

segmentation. While InceptionResNetV2 produced the most visually refined results, its computational complexity makes it less suitable for real-time deployment. In contrast, ResNet18 provided visually comparable masks, with segmentation quality not significantly inferior to that of Vanilla U-Net, despite being much lighter.

3.3 Impact of the triplet setup

To assess whether the proposed triplet input improves temporal stability compared to single-frame predictions, we trained a ResNet18 model with both single-frame and triplet inputs under otherwise identical settings. We then defined temporal consistency metrics, following recent studies [33, 34], and reported them in Table 2. *Temporal Dice* is defined as the Dice coefficient between consecutive frame predictions, averaged over the sequence, while *Temporal IoU* is defined analogously using the IoU.

During training, the model predicts the segmentation of the central frame in each temporal triplet, which is a common setup allowing the network to learn temporal context from both future and past frames. During real-time inference, the future frame is not available, and the model therefore operates on the current frame and its two earlier frames. This creates a small mismatch between training and inference conditions and remains a limitation of the present implementation. A future extension could use only past frames during training to fully align with real-time constraints.

In addition, we evaluated prediction stability across time. Following Rebol et al. [35], we measured *Flip-rate,* the proportion of pixels whose labels switch between consecutive frames (e.g., a vessel pixel that disappears and reappears). This captures segmentation "flickering" over time. However, in our experiments, Flip-rate values were consistently close to zero, likely reflecting both the strong class imbalance (vessel vs. background) and the generally high segmentation accuracy. Thus, the Flip-rate confirms the absence of large temporal instabilities in both compared models.

To quantify boundary stability, we used the idea from Perazzi et al. [36]. We define *Boundary jitter* as the average displacement (in pixels) of the predicted vessel boundary between consecutive frames, capturing small shifts of vessel contours. The results indicate that our model performs better when trained with adjacent frames, enhancing all temporal metrics. We also assessed the per-frame Dice score against the ground truth (GT), which showed slight improvement. Additionally, the triplet input configuration improved temporal consistency. It produced masks that were more stable and less prone to flickering compared to the single-frame predictions shown in Figure 4. The triplet-frame setup also reduced both false positive and false negative predictions. Overall, it generated smoother masks and a more stable stream during real-time experiments.

Beyond the single-frame versus triplet comparison, we also conducted a follow-up experiment to examine how different temporal input configurations influence model behavior. We compared single-frame, triplet, five-frame, seven-frame, and a far-frames (frames at positions: −3, 0, +3), all under identical training conditions. All temporal configurations performed better than the single-frame baseline in terms of training convergence and validation accuracy, confirming that temporal context is beneficial for LUS vessel segmentation. Full training curves for this ablation are provided in Appendix A. Increasing the number of adjacent frames (five or seven) did not yield additional gains, suggesting that neighbouring frames are highly redundant. Interestingly, the far-frame setup, which introduces broader temporal separation, produced the most stable validation loss and the highest validation Dice. This suggests that temporally distant frames provide more useful information than tightly clustered ones. Such observation may be worth examining further to understand how different temporal sampling strategies affect model performance. Overall, the results support the use of temporal inputs and show that the triplet configuration offers a practical balance between accuracy, stability, and real-time feasibility. However, these trends are specific to our dataset and acquisition setup, and different probe motion dynamics or frame sampling rates may lead to different temporal dependencies.

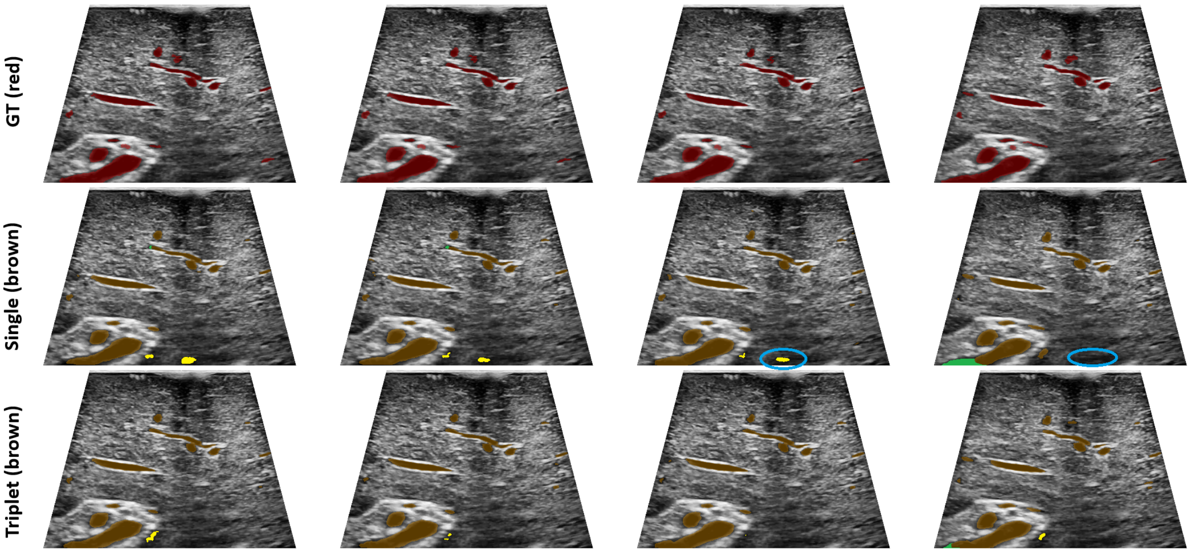

Figure 4. Qualitative comparison of vessel segmentation results from single-frame and triplet-based ResNet18 U-Net models. Green color highlights false negatives (under-predicted vessels), while yellow color marks false positives (over-predicted regions), and blue circles indicate temporal flipping across frames.

3.4 Real-time experiment and results

To test deployment under realistic conditions, the models were evaluated on an unseen laparoscopic ultrasound video using the full real-time pipeline, including preprocessing, frame-wise inference, and post-processing to mimic the clinician's view. The developed dynamic contour detector enabled the model to adapt to changing imaging depths without introducing artifacts, maintaining high frames per second (fps) even with intensive preprocessing and post-processing.

The real-time performance metrics of the tested models are summarized in Table 3. The final column reflects the results observed during continuous video playback in fps. Despite InceptionResNetV2 achieving the highest Dice score, it was the slowest for inference. The literature suggests that a frame rate of 30 fps is generally sufficient for real-time performance [37, 38]. A couple of tested models, including ResNet18, ResNet50, and MobileNet_v2 reached and sustained this fps target, with ResNet18 also allowing for potential further tuning if needed. Notably, this model delivered segmentation quality comparable to that of InceptionResNetV2 while achieving a stable throughput of 40 fps, making it the most viable candidate for intraoperative segmentation deployment.

Table 3. Comparison of model complexity and performance. Parameter count, forward-pass inference time, model size, and measured real-time fps (including preprocessing and postprocessing in the ultrasound pipeline) are reported.

Encoder name	Parameters	Time (ms)	Model size (MB)	FPS
ResNet18	14,328,209	**5.48**	56.08	**40**
ResNet50	32,521,105	9.03	127.38	36
MobileNet_v2	6,628,945	7.55	26.17	37
Vanilla U-Net	31,037,633	12.63	121.33	29
DenseNet121	13,607,633	16.97	53.78	29
IncResNetV2	62,029,297	29.21	243.04	21

3.5 Dataset limitations

The dataset used in this study was relatively small, consisting of 11 acquisitions and 2,200 images, however it is comparable to the dataset sizes reported in other liver vessel segmentation studies [6, 10]. We recognize, that limited number of acquisitions constrains the amount of independent data available for validation. As we kept one test set for final testing, the remaining ten acquisitions were used for 5-fold CV and model development. Despite this limitation, the variation between folds in the 5-fold CV results, presented in Table 1, remained low. This indicated a stable model performance. In the future, other evaluation techniques such as 3-fold cross-validation or leave-one-out validation could be explored. Finally, if more annotated LUS data become available, the model can be retrained to further strengthen generalizability.

3.6 Model selection limitations

CNN-based backbones, used in this study, are not the newest architectures, but they remain widely applied and competitive in medical image segmentation. Other types, like transformer based segmentation models represent a more recent research direction, but they typically require much larger datasets and higher computational resources, which were not available in this study. Given our focus on developing an accurate and real-time vessel segmentation model suitable for intraoperative use, we prioritized architectures that offer strong performance with limited data and efficient inference speed. For these reasons, transformer based methods were not further explored, but they remain a potential extension if more annotated data become available.

3.7 Dataset characteristics and clinical context

The dataset consists intraoperative ultrasound sequences collected from patients with various underlying liver conditions, including fatty liver and cirrhosis, which contribute to variability in image appearance. During the annotation process, the reviewing radiologist noted several cases showing typical features, including signs of fatty liver or cirrhotic change, as well as cysts, tumors, or marks from prior ablation procedures when visible within the 200 frame acquisitions. This variability reflects the real world conditions under which LUS vessel segmentation models must operate. While the dataset includes a wide spectrum of liver pathologies, the present study did not perform a pathology specific analysis of model performance. Such an investigation would be a valuable extension for future work.

3.8 Multimodal extension

We also considered whether the method could be extended to a multimodal setting by, for example, incorporating Doppler information, which highlights vascular structures through flow and velocity patterns. Prior work, such as Jiang et al. [39], has shown that combining Doppler with B-mode data can improve vascular segmentation by providing complementary physiological information. However, our retrospective LUS dataset contained only B-mode recordings and did not include any Doppler channels, making such multimodal fusion currently infeasible. Exploring Doppler-augmented segmentation therefore remains an interesting direction for future research.

4 Conclusion

In this paper, we presented a five-step preprocessing framework combined with a triplet-based ResNet18 U-Net model for real-time laparoscopic ultrasound image segmentation, achieving competitive Dice scores for liver vessel segmentation. Key contributions include a dynamic contour detector that improved generalization across varying depths and a triplet input setup that enhanced the temporal stability of vessel segmentation. We also evaluated real-time performance and mask quality, confirming the feasibility of deployment. Future work will focus on direct integration with live ultrasound streams and extension to 3D vessel reconstruction.

Acknowledgments

The authors acknowledge the guidance and support of Rafael Palomar. This study was carried out as part of the EU project HoloSurge, project number 101137233, which aims to develop a multi-modal 3D holographic tool and real-time guidance system.

References

[1] The World Cancer Research Fund. *Liver cancer statistics.* 2022. URL: https://www.wcrf.org/preventing-cancer/cancer-statistics/liver-cancer-statistics/.

[2] E. P. Weledji, G. E. Orock, M. N. Ngowe, and D. S. Nsagha. “How grim is hepatocellular carcinoma?” In: *Annals of Medicine and Surgery* 3.3 (2014). ISSN: 2049-0801. DOI: 10.1016/j.amsu.2014.06.006.

[3] E. Kazam. “Ultrasound Teaching Manual: The Basics of Performing and Interpreting Ultrasound Scans”. In: *Clinical Imaging* 23.6 (Nov. 1999). ISSN: 0899-7071. DOI: 10.1016/S0899-7071(99)00133-3.

[4] M. Baad, Z. F. Lu, I. Reiser, and D. Paushter. “Clinical Significance of US Artifacts”. In: *RadioGraphics* 37.5 (Aug. 2017). ISSN: 0271-5333. DOI: 10.1148/rg.2017160175.

[5] J. F. Gerstenmaier and R. N. Gibson. “Ultrasound in chronic liver disease”. In: *Insights Imaging* 5.4 (2014). ISSN: 18694101. DOI: 10.1007/s13244-014-0336-2.

[6] M. Y. Ansari, Y. Yang, P. K. Meher, and S. P. Dakua. “Dense-PSP-UNet: A neural network for fast inference liver ultrasound segmentation”. In: *Computers in Biology and Medicine* 153 (2023), p. 106478. ISSN: 0010-4825. DOI: https://doi.org/10.1016/j.compbiomed.2022.106478.

[7] P. R. Hoskins. “Principles of doppler ultrasound”. In: *Diagnostic Ultrasound, Third Ed. Phys. Equip.* (2019), pp. 143–158. DOI: 10.1201/b14901-8.

[8] A. N. Abou Ali, A. Fittipaldi, J. Rocha-Neves, B. Ruaro, F. Benedetto, Z. Al Ghadban, G. Simon, S. Lepidi, and M. D’Oria. “Clinical applications of contrast-enhanced ultrasound in vascular surgery: State-of-the-art narrative and pictorial review”. In: *JVS-Vascular Insights: An Open Access Publication from the Society for Vascular Surgery* 3 (Jan. 2025). ISSN: 2949-9127. DOI: 10.1016/j.jvsvi.2025.100254.

[9] D. L. Pham, C. Xu, and J. L. Prince. “Current methods in medical image segmentation.” In: *Annual review of biomedical engineering* 2 (2000), pp. 315–37. DOI: 10.1146/annurev.bioeng.2.1.315.

[10] K. Tanaka, T. Kurihara, Y. Takahashi, S. Onogi, T. Sugino, Y. Nakajima, Y. Edamoto, and K. Masuda. “Segmentation of Liver Blood Vessel in Ultrasound Images Using Mask R-CNN”. In: *Advanced Biomedical Engineering* 13 (2024), pp. 379–388. DOI: 10.14326/abe.13.379.

[11] J. Zeng, D. Jha, E. Aktas, E. Keles, A. Medetalibeyoglu, M. Antalek, R. Lewandowski, D. Ladner, A. A. Borhani, G. Durak, and U. Bagci. *A Reverse Mamba Attention Network for Pathological Liver Segmentation.* 2025. DOI: 10.48550/arXiv.2502.18232.

[12] H. B. Jenssen, V. Nainamalai, E. Pelanis, R. P. Kumar, A. Abildgaard, F. K. Kolrud, B. Edwin, J. Jiang, J. Vettukattil, O. J. Elle, and Å. s. A. Fretland. “Challenges and artificial intelligence solutions for clinically optimal hepatic venous vessel segmentation”. In: *Biomedical Signal Processing and Control* 106 (2025), p. 107822. ISSN: 1746-8094. DOI: 10.1016/j.bspc.2025.107822.

[13] 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.* Ed. by N. Navab, J. Hornegger, W. M. Wells, and A. F. Frangi. Springer International Publishing, 2015, pp. 234–241. DOI: 10.1007/978-3-319-24574-4_28.

[14] M. Regalado, N. Carreras, C. Mata, A. Oliver, X. Lladó, and T. Agut. “Automatic Segmentation of Sylvian Fissure in Brain Ultrasound Images of Pre-Term Infants Using Deep Learning Models”. In: *Ultrasound in Medicine & Biology* 3 (2025), pp. 543–550. ISSN: 0301-5629. DOI: 10.1016/j.ultrasmedbio.2024.11.016.

[15] A. Sulaiman, V. Anand, S. Gupta, A. Rajab, H. Alshahrani, M. S. Al Reshan, A. Shaikh, and M. Hamdi. “Attention based UNet model for breast cancer segmentation using BUSI dataset”. In: *Scientific Reports* 14.1 (2024), p. 22422. ISSN: 2045-2322. DOI: 10.1038/s41598-024-72712-5.

[16] Y. Wan, Y. Yang, H. Guo, Y. Yan, T. Liu, W. Liu, Y. Wang, W. Wang, and H. Dang. “D-TransUNet: A Breast Tumor Ultrasound Image Segmentation Model Based on Deep Feature Fusion”. In: *Journal of Artificial Intelligence for Medical Sciences* 5.1–2 (2024), pp. 1–8. DOI: 10.55578/joaims.240226.001.

[17] M. Awais, M. Al Taie, C. S. O’Connor, A. H. Castelo, B. Acidi, H. S. Tran Cao, and K. K. Brock. “Enhancing Surgical Guidance: Deep Learning-Based Liver Vessel Segmentation in Real-Time Ultrasound Video Frames”. In: *Cancers* 16.21 (2024), p. 3674. ISSN: 2072-6694. DOI: 10.3390/cancers16213674.

[18] E. Smistad, T. Lie, and K. F. Johansen. “Real-time segmentation of blood vessels, nerves and bone in ultrasound-guided regional anesthesia using deep learning”. In: *2021 IEEE International Ultrasonics Symposium (IUS).* 2021, pp. 1–4. DOI: 10.1109/IUS52206.2021.9593525.

[19] J. Ramalhinho, B. Koo, N. Montaña-Brown, S. U. Saeed, E. Bonmati, K. Gurusamy, S. P. Pereira, B. Davidson, Y. Hu, and M. J. Clarkson. “Deep hashing for global registration of untracked 2D laparoscopic ultrasound to CT.” In: *International journal of computer assisted radiology and surgery* 17 (2022), pp. 1461–1468. DOI: 10.1007/s11548-022-02605-3.

[20] H. D. Le, M. Q. Vu, M. T. Tran, and N. Van Phuc. “Triplet Temporal-based Video Recognition with Multiview for Temporal Action Localization”. In: *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW).* 2023, pp. 5428–5434. DOI: 10.1109/CVPRW59228.2023.00573.

[21] A. M. Reza. “Realization of the Contrast Limited Adaptive Histogram Equalization (CLAHE) for Real-Time Image Enhancement”. In: *Journal of VLSI signal processing systems for signal, image and video technology* 38.1 (2004), pp. 35–44. ISSN: 0922-5773. DOI: 10.1023/B:VLSI.0000028532.53893.82.

[22] L. Wu, Y. Ling, L. Lan, K. He, C. Yu, Z. Zhou, and L. Shen. “Automatic Segmentation of the Left Ventricle in Apical Four-Chamber View on Transesophageal Echocardiography Based on UNeXt Deep Neural Network”. In: *Diagnostics* 14.23 (2024), p. 2766. ISSN: 2075-4418. DOI: 10.3390/diagnostics14232766.

[23] K. He, X. Zhang, S. Ren, and J. Sun. “Deep Residual Learning for Image Recognition”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).* 2016, pp. 770–778. DOI: 10.1109/CVPR.2016.90.

[24] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. “ MobileNetV2: Inverted Residuals and Linear Bottlenecks ”. In: *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR).* IEEE Computer Society, 2018, pp. 4510–4520. DOI: 10.1109/CVPR.2018.00474.

[25] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. “Densely Connected Convolutional Networks”. In: *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2017, pp. 2261–2269. DOI: 10.1109/CVPR.2017.243.

[26] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna. “Rethinking the Inception Architecture for Computer Vision”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE Computer Society, 2016, pp. 2818–2826. DOI: 10.1109/CVPR.2016.308.

[27] P. Iakubovskii. *Segmentation Models Pytorch*. https://github.com/qubvel/segmentation_models.pytorch. 2019.

[28] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. “ImageNet: A large-scale hierarchical image database”. In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. 2009, pp. 248–255. DOI: 10.1109/CVPR.2009.5206848.

[29] 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)*. 2016, pp. 565–571. DOI: 10.1109/3DV.2016.79.

[30] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. “Focal Loss for Dense Object Detection”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42.2 (2020), pp. 318–327. DOI: 10.1109/TPAMI.2018.2858826.

[31] I. G. Courville, Y. Bengio, and Aaron. *Deep Learning*. MIT Press, 2016, Chapter 6: Deep Feedforward Networks. URL: http://www.deeplearningbook.org.

[32] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala. “PyTorch: An imperative style, high-performance deep learning library”. In: *Advances in Neural Information Processing Systems* 32 (2019). ISSN: 10495258. DOI: 10.48550/arXiv.1912.01703.

[33] S. Varghese, Y. Bayzidi, A. Bär, N. Kapoor, S. Lahiri, J. D. Schneider, N. Schmidt, P. Schlicht, F. Hüger, and T. Fingscheidt. “Unsupervised Temporal Consistency Metric for Video Segmentation in Highly-Automated Driving”. In: *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*. 2020, pp. 1369–1378. DOI: 10.1109/CVPRW50498.2020.00176.

[34] H. Wei, J. Ma, Y. Zhou, W. Xue, and D. Ni. “Co-learning of appearance and shape for precise ejection fraction estimation from echocardiographic sequences.” eng. In: *Medical image analysis* 84 (Feb. 2023), p. 102686. DOI: 10.1016/j.media.2022.102686.

[35] M. Rebol and P. Knöbelreiter. “Frame-To-Frame Consistent Semantic Segmentation”. In: *Proceedings of the OAGM/AAPR Workshop 2020*. 2020. DOI: 10.3217/978-3-85125-752-6-18.

[36] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. “A Benchmark Dataset and Evaluation Methodology for Video Object Segmentation”. In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016, pp. 724–732. DOI: 10.1109/CVPR.2016.85.

[37] S. Vaze, W. Xie, and A. I. L. Namburete. “Low-Memory CNNs Enabling Real-Time Ultrasound Segmentation Towards Mobile Deployment”. In: *IEEE Journal of Biomedical and Health Informatics* 24.4 (2020), pp. 1059–1069. DOI: 10.1109/JBHI.2019.2961264.

[38] T. Natali, A. Zhylka, K. Olthof, J. N. Smit, T. R. Baetens, N. F. M. Kok, K. F. D. Kuhlmann, O. Ivashchenko, T. J. M. Ruers, and M. Fusaglia. “Automatic hepatic tumor segmentation in intra-operative ultrasound: a supervised deep-learning approach.” In: *Journal of medical imaging (Bellingham, Wash.)* 11 (2 Mar. 2024), p. 024501. DOI: 10.1117/1.JMI.11.2.024501.

[39] B. Jiang, A. Chen, S. Bharat, and M. Zheng. “Automatic Ultrasound Vessel Segmentation with Deep Spatiotemporal Context Learning”. In: *Simplifying Medical Ultrasound*. Springer International Publishing, 2021, pp. 3–13. DOI: 10.1007/978-3-030-87583-1_1.

Appendix A. Brief Analysis of Temporal Frame Configurations

To investigate the effect of different temporal input configurations, all models were trained on the same patient split, with Case 8 held out as the fixed test set and excluded from this experiment. The remaining subjects were used for training and validation, and early stopping was applied based on the validation macro Dice. Figures (A.1–A.3) summarize training loss, validation loss, and validation Dice across epochs for all temporal setups.

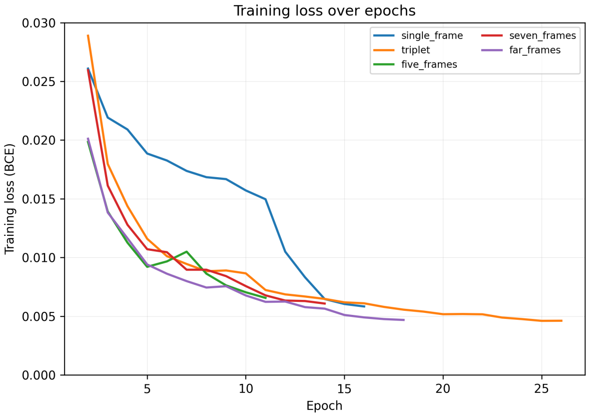

Figure A.1. Smoothened training loss over epochs for all temporal input configurations, using early-stopping.

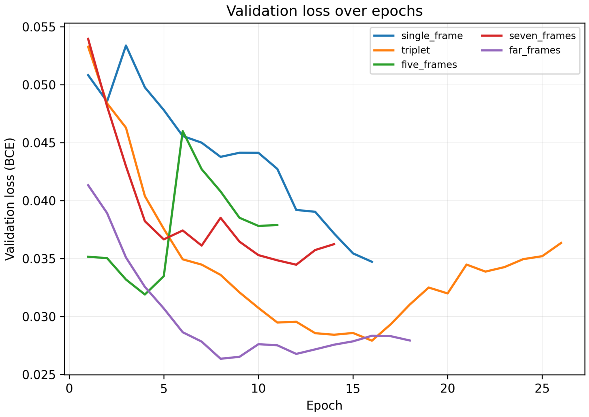

Figure A.2. Smoothened validation loss over epochs for all temporal input configurations, using early-stopping.

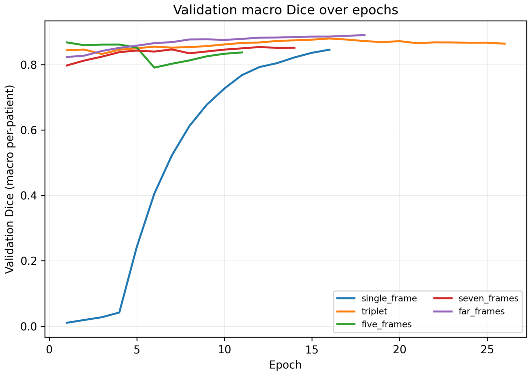

Figure A.3. Smoothened validation macro Dice across epochs for different temporal input configurations, using early-stopping.