CS482/682 Final Project Report Group 2

Robot Tool Segmentation

Seyi R. Afolayan (oafolay2), Sameer Khan (skhan171),
Kedar Krishnan (kkrish13), Kent Shibata (kshibat2)

1 Introduction

Background The goal of this project is to accurately and reliably segment and identify robotic tools in laparoscopic surgery, equipped to the da Vinci robot system. Segmentation of such tools is difficult, due to visual noises such as reflection, water mist, motion blur, etc. Precise tool segmentation is crucial for automating surgical tasks, as it facilitates real-time visual-servoing of robotic joints. This project aims to train and evaluate a model capable of robust tool segmentation in diverse and challenging surgical scenarios, using the Endovis2018 challenge dataset. The focus will be to recreate the BBANet [1] model that handles visual noise, delivering high segmentation accuracy to support autonomous/semi-autonomous surgical applications.

Related Work Medical tool/robot segmentation has been explored by many groups in supervised and unsupervised methods. U-Net [2], one of the earliest and most famous segmentation model, uses a U-shaped CNN architecture for segmentation. TransUNet [3] builds upon the U-Net architecture by leveraging the features created by the U-Net convolution layers as tokenized inputs to a visual transformer, which learns the segmentation. We decided to recreate BBANet [1], a multistage approach to segmentation. It uses a pre-trained CNN as the encoder, then feeds into a multi-scale fusion block, and finally inputs these into a sequence of dual branch attention transformers. We based our model on the structure of BAANet with some enhancements for faster training.

2 Methods

Dataset To train and test our model, we used the Endovis2018 dataset, a part of the 2018 Robotic Scene Segmentation Challenge [4]. This was collected with the da Vinci Xi, and features 15 training video sequences, 4 testing sequences, and 12 different labels, 6 of which relate to robotic tools. We further split the training sequences into 11 training scenes and 4 validation sequences, 2,5,9 and 15, for hyperparameter tuning. The original images and labels have a resolution of 1280 x 1024 pixels, making them large for our compute systems, so we chose to downsample them by 4 to aid training on Google Colaboratory. Along with this, simple data augmentations such as flips, color augmentation, and normalization were applied to make our training more robust to variances in orientation and lighting.

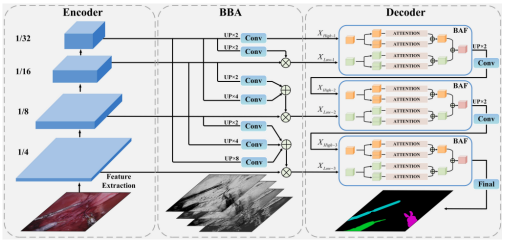

Figure 1: Branch Aggregation Attention Network model architecture [1]

Model, Training and Evaluation For our model, we reconstructed the BAANet by W. Shen et al. [1] shown in Figure 1. We chose this model because of its relatively novel approach, consisting of

modules that were easily interpretable and familiar. BAANet consists of 3 sections: Encoder, Branch Balance Aggregation (BBA), and Block Attention Fusion (BAF). We used a pre-trained MobileNetv2 [5] as the encoder to generate input feature maps at various scales. These feature maps are fed into the BBA module, combining high and low-level feature maps using element-wise addition and multiplication. The addition ensures robust feature fusion to integrate higher-level semantic information with lower-level spatial details. Multiplication balances the influence of these features to suppress the noise of surgical conditions.

These new maps are then processed through dual attention blocks, each with spatial and channel-wise attention. Spatial attention refines local information on instrument locations, while channel-wise attention emphasizes important features such as edges and textures that are critical to accurate segmentation. To understand the importance of these modules, we performed an ablation study with three models: BAANet, BAANet with only the BBA module, and BAANet with only the BAF module. This gave us a better understanding of the unique contributions of both modules to overall performance.

We trained each model for 50 epochs using DICE loss for consistency and time efficiency, with a learning rate of 1e-4 in agreement with an ADAM optimizer. Due to resource constraints, we trained a single instance of each model. Evaluation metrics included the mean DICE and IoU scores, as well as their respective values for robotic tool labels only.

3 Results

Table 1: Mean Test Data Scores for Three Models on all Labels and Robot Labels

Scores	BAANet	BBA Only	BAF Only
mDICE	0.672	0.685	0.493
mIoU	0.633	0.638	0.476
mRDICE	0.704	0.688	0.010
mRIoU	0.605	0.591	0.098

After training for 50 epochs, we obtained the training loss and validation accuracy curves in Figure 2. In the test set, we obtained the DICE and IoU values shown in Table 1, showing that BAAnet and BBA only performed similarly well on all classes, but BAANet did slightly better on robot classes.

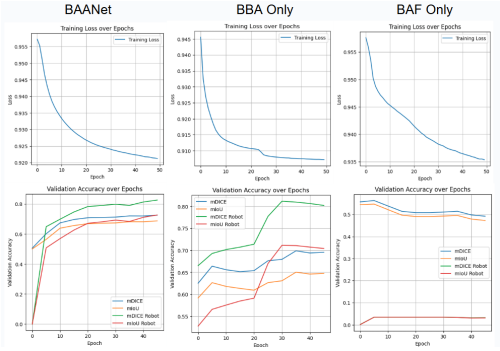

Figure 2: Training Loss and Validation Scores for 1. BAANet, 2. BBA only, 3. BAF only

4 Discussion

The results highlight the limitations of the BAF module, which achieves an overall DICE score (0.493), but a much lower robot tool DICE score (0.010). This suggests that the BAF module alone struggles to capture finer details and specific tool regions due to the lack of multi-scale feature fusion and noise suppression provided by the BBA module. In contrast, the BBA module achieved higher DICE scores of 0.672 and 0.704, though it showed signs of overfitting. Meanwhile, the BAF and complete BAANet models showed no sign of convergence, indicating that the BAF module may enhance the BBA module's accuracy with further training.

The most challenging factors for training include the high number of labels, the downsampling of the inputs, and the high training time requirement. The large number of labels make the learning more complex while downsampling the images gives less information to the network. With better computational resources and better optimized training parameters, BAANet would perform much better.

References

[1] W. Shen, Y. Wang, M. Liu, J. Wang, R. Ding, Z. Zhang, and E. Meijering, "Branch aggregation attention network for robotic surgical instrument segmentation," *IEEE Transactions on Medical Imaging*, vol. 42, no. 11, pp. 3408–3419, 2023.

[2] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

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

[4] M. Allan, S. Kondo, S. Bodenstedt, S. Leger, R. Kadkhodamohammadi, I. Luengo, F. Fuentes, E. Flouty, A. Mohammed, M. Pedersen, A. Kori, V. Alex, G. Krishnamurthi, D. Rauber, R. Mendel, C. Palm, S. Bano, G. Saibro, C.-S. Shih, H.-A. Chiang, J. Zhuang, J. Yang, V. Iglovikov, A. Dobrenkii, M. Reddiboina, A. Reddy, X. Liu, C. Gao, M. Unberath, M. Kim, C. Kim, C. Kim, H. Kim, G. Lee, I. Ullah, M. Luna, S. H. Park, M. Azizian, D. Stoyanov, L. Maier-Hein, and S. Speidel, "2018 robotic scene segmentation challenge," 2020.

[5] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "Mobilenetv2: Inverted residuals and linear bottlenecks," 2019.