VIT Hackathon 2024 –
Problem Statement 03: Surgical Tool Segmentation

Background

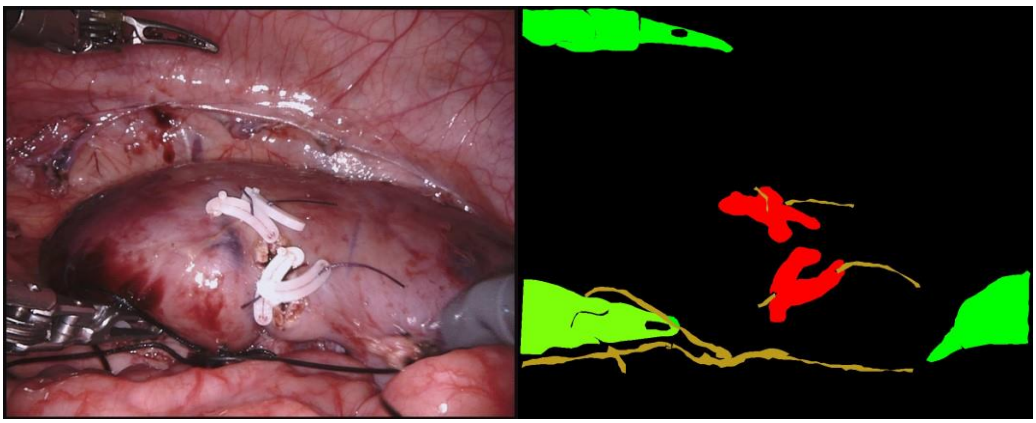
The segmentation of robotic instruments presents a significant challenge for roboticassisted minimally invasive surgery. This challenge includes a spectrum of applications, ranging from basic 2D functions like overlay masking and 2D tracking to more intricate 3D features such as pose estimation. Accurate segmentation and understanding of surgical videos has a great impact in improving automated surgeries and guidance for surgeons . Which, in turn can significantly improve surgery's outcomes and patients safety.

Are you ready to make healthcare safer and better ?

In this hackathon we invite the participants to engage into the following tasks : Binary segmentation and tool recognition. Binary segmentation entails the separation of the image into two categories: instruments and background. Meanwhile, instrument recognition consists in recognizing which type of tool specifically to which each binary segmentaion belongs. The task can be completed using a single deep learning model.

Task overview

Participants are tasked with developing deep learning models that can perform the task of segmenting and recognizing the instruments shown in the videos. In each frame, a variety of surgical tools are present (Instruments , Clamps, threads) . These tools need to be segmented according to their category along with the tissue appearing in the background.



Dataset

The training dataset is made up of 16 robotic procedures. The original video data was recorded at 60 Hz and to reduce labelling cost they have been subsampled to 2 Hz. Sequences with little or no motion are manually removed to leave 149 frames per procedure. Video frames are 1280x1024 for which labeling is provided.

The classes found in the training and test will be:

- Instrument
- Drop in Ultrasound Probe
- Suturing Needles
- Suturing thread
- Clips/clamps
- Background tissue

Each class will have a distinct numerical label in a ground truth image. A supplied json file will contain the class name to numerical label mapping.

Submission Guidelines

- Participants should submit their trained models along with inference code.
- Participants should upload all the code into a GitHub repository. The submitted code should be able to run on our end without any changes other than a simple path change.
- Participants should submit a pdf along with the code outlining the detailed steps to run the code and the thought process and approach taken in detail to solve the problem statement.
- Results will be evaluated on a separate test set (not provided during the hackathon).
- Teams can use any deep learning framework (e.g., PyTorch, TensorFlow).
- In case a Bonus Challenge is attempted it should be mentioned along with all its details.

Evaluation Metrics

- **Jaccard Index (IoU):** Measure the overlap between predicted and ground-truth object masks.
- **F1-score:** Balance precision and recall.

Bonus Challenges **(Optional additional points for successful completion)**

- **Real-Time Inference:** Optimize your models for real-time performance.
- **Temporal Consistency:** Improve tracking across frames by considering temporal context.