

# Towards Surgical Context Inference and Translation to Gestures

Kay Hutchinson<sup>1\*</sup>, Zongyu Li<sup>1\*</sup>, Ian Reyes<sup>2\*</sup>, Homa Alemzadeh<sup>1</sup>

**Abstract**—Manual labeling of gestures in robot-assisted surgery is labor intensive, prone to errors, and requires expertise or training. We propose a method for automated and explainable generation of gesture transcripts that leverages the abundance of data for image segmentation. Surgical context is detected using segmentation masks by examining the distances and intersections between the tools and objects. Next, context labels are translated into gesture transcripts using knowledge-based Finite State Machine (FSM) and data-driven Long Short Term Memory (LSTM) models. We evaluate the performance of each stage of our method by comparing the results with the ground truth segmentation masks, the consensus context labels, and the gesture labels in the JIGSAWS dataset. Our results show that our segmentation models achieve state-of-the-art performance in recognizing needle and thread in Suturing and we can automatically detect important surgical states with high agreement with crowd-sourced labels (e.g., contact between graspers and objects in Suturing). We also find that the FSM models are more robust to poor segmentation and labeling performance than LSTMs. Our proposed method can significantly shorten the gesture labeling process ( $\sim 2.8$  times).

## I. INTRODUCTION

Surgical robots for minimally invasive surgery (MIS) enable surgeons to operate with greater flexibility and precision, thus reducing incision size, recovery time, and scarring. Their widespread adoption into surgical specialties such as urology, gynecology, and general surgery has opened up new fields of interdisciplinary research. Gesture segmentation and classification has been one of those research areas where both supervised [1], [2], [3], [4], [5], [6] and unsupervised learning [7], [8], [9], [10], [11] approaches have been developed for gesture recognition. However, these approaches either rely on black-box deep learning models that are hard to verify and need extensive training data or do not capture the human interpretable contextual information of the gestures.

The JIGSAWS dataset [12] with its surgical gesture labels has been the foundation of many advancements in surgical gesture recognition [13], surgical process modeling [1], skill assessment [14], [15], error detection [16], [17], and autonomy [18]. However, unlike annotations for surgical instrument segmentation, annotations for surgical workflow such as gestures need guidance from surgeons [19]. Labeling using descriptive gesture definitions is tedious and subjective, leaving uncertainty as to exactly when gestures start and end, and can have annotation errors that can adversely impact

machine learning models and analyses [13], [20]. Recent studies using the JIGSAWS dataset have found errors in  $\sim 2$ -10% of the gesture labels [21], [20]. As emphasized in [13], larger labeled datasets using a common surgical language are needed to support collaboration and comparative analysis.

Some recent works have focused on finer-grained surgical actions such as action triplets [22], [23], [24] and motion primitives [25] based on the interactions between robotic tools and objects in the surgical environment. [25] presented a formal framework for modeling surgical tasks with a unified set of motion primitives that cause changes in surgical context captured from the physical environment. These motion primitives were shown to be generalizable across different surgical tasks and can be used to combine data from different datasets. [25] suggests a relation between context and existing gesture labels, but does not define direct relations between the two.

Furthermore, despite limited availability of datasets that include kinematic data from surgical robots, datasets for instrument and object segmentation in MIS procedures are plentiful and have been the subject of imaging competitions [26], [27]. We propose methods that leverage the abundance of data with image annotations for surgical instruments and important surgical objects to address the challenges of manual labeling and relate surgical context to gestures. Our goal is to develop an automated, independent, and explainable way of generating gesture transcripts based on video data that does not rely on expensive training data on gestures. Such a method would be easier to verify by humans/experts and can be used as the ground truth for evaluating the black-box gesture recognition models that directly detect gestures from kinematic data.

The main contributions of the paper are as follows:

- We present a method for the automated inference of surgical context based on detecting important surgical tool and object interactions using image segmentation.
- We propose two methods for automated translation of context labels to gesture labels based on a knowledge-based finite state machine model and a data-driven machine learning model.
- We use the JIGSAWS dataset as a case study to demonstrate that our proposed approach results in shorter labeling time using the segmentation masks.

## II. PRELIMINARIES

### A. Surgical Process Modeling

Surgical process modeling [28] defines how surgical procedures can be decomposed into steps, tasks, and gestures

\* denotes equal contribution.

<sup>1</sup> Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22903 USA. <sup>2</sup> Ian Reyes was with the Department of Computer Science, University of Virginia, Charlottesville, VA 22903 USA. He is now with IBM. {kch4fk, z17qw, ir6mp, ha4d}@virginia.edu Code for this paper is available at [https://github.com/UVA-DSA/Auto\\_Surgical\\_Context2Gesture](https://github.com/UVA-DSA/Auto_Surgical_Context2Gesture)

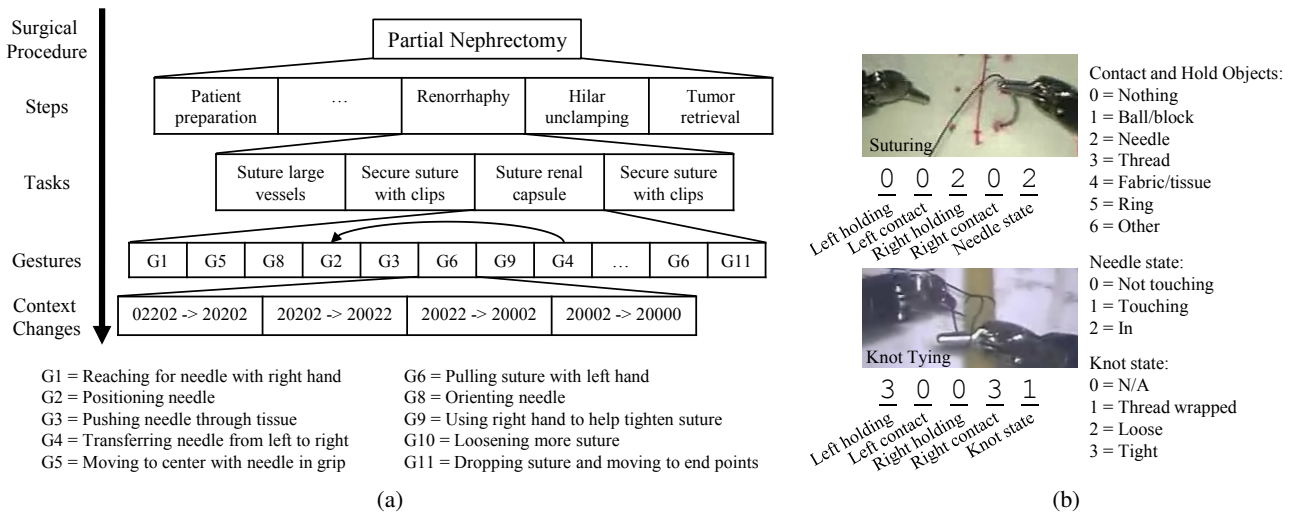


Fig. 1: 1a Surgical hierarchy and relation between gestures and context in a suturing task. 1b State variables and object encodings that comprise context for the JIGSAWS tasks (see Figure 2). In the Suturing and Needle Passing tasks, a needle is used to throw four sutures through the fabric and rings, respectively, while two knots are tied in the Knot Tying task.

as shown in Figure 1a. Gestures are defined as actions with semantic meaning for a specific intent and involve particular tools and objects. Thus, they explicitly include the surgical context, capturing important states and interactions in the physical environment. The formal framework in [25] extended this hierarchy to further include the finer-grained motion primitives (or verbs in action triplets [24], [29]) as the atomic units of surgical activity (e.g., grasp, push) that lead to changes in context, without explicitly including the semantics of physical context (e.g. needle through tissue).

### B. Surgical Context

Surgical context is defined as a set of state variables describing the status of a task and interactions among the surgical instruments, objects, and anatomical structures in the physical environment [30], [16], [25]. As shown in Figure 1b, the first four state variables represent objects held by or in contact with the surgical instruments and are the general state variables for all tasks. The fifth state variable is task-specific and represents task progress; i.e., the needle’s relation to the fabric or ring in the Suturing and Needle Passing tasks, or the knot’s status in the Knot Tying task. Figure 1b shows the general and task-specific state variables with their possible values in the Suturing and Knot Tying tasks of the JIGSAWS dataset. In Figure 1b, the example context of 00202 in the Suturing task means that the right grasper is holding the needle and the needle is in the fabric.

The COMPASS dataset [25] has context labels for all three tasks in the JIGSAWS dataset based on consensus among three annotators. But, it does not provide translations from context or motion primitives to gestures which limits comparisons to existing works. Manual labeling was needed to create the context labels which is still subjective and time consuming, despite achieving near-perfect agreement with expert surgeons. However, [25] showed that high quality surgical workflow labels can be generated by examining state variables that comprise the context. With recent improve-

ments in surgical scene segmentation, we show that context can be detected automatically from video data.

### C. Surgical Scene Segmentation

To advance analysis on video data and provide insights on surgeon performance, the 2017 and 2018 EndoVis workshops at MICCAI introduced a challenge to perform robotic instrument and scene segmentation using images from a da Vinci Xi robot in porcine procedures [26]. Various models have been proposed in the challenge, but segmenting all objects in a surgical scene has been challenging. The DeepLab V3+ model [31] achieved the best overall performance in [27] (see Table I). Other DeepLab models [32], [31] have also shown promise in surgical tool and object segmentation.

Most existing works on robot instrument or surgical scene segmentation were based on real surgery videos using publicly available datasets such as MICCAI EndoVis 17 [26], MICCAI EndoVis 18 [27] and Cata7 [33]. Popular frameworks include UNet [34], TeraNet [35], and LinkNet [36]. Surgical scene segmentation in the dry-lab settings with the JIGSAWS dataset was done in [37] and [38], but we go further by segmenting additional objects and using tool and object segmentation for context inference. Although surgical scene segmentation and instrument tracking can be used for skill assessment [39], they have not yet been used for automatic context and gesture inference. Hence, our approach could be used as an independent source to evaluate context or gesture segmentation models trained using kinematic data.

Further, we aim to integrate data-driven segmentation with knowledge-driven context inference and context to gesture translation to perform gesture recognition. Compared to the above deep learning approaches for gesture recognition, this approach enables improvements by integrating human input. Our method also benefits from the availability of large open source image segmentation datasets that provide pretrained weights for segmentation models and could also improve segmentation performance via fine-tuning on smaller datasets.

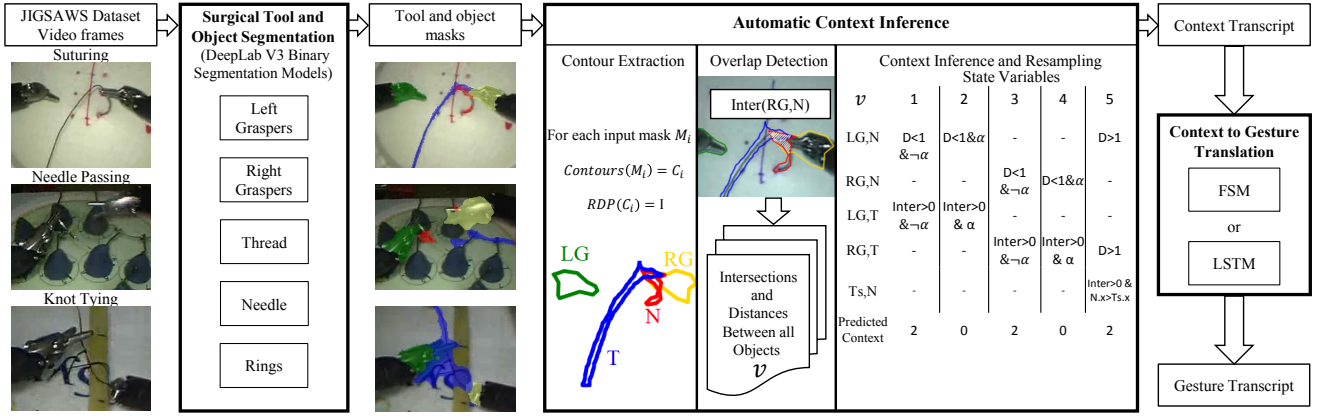


Fig. 2: Pipeline for automatic context inference based on segmentation of video data and context to gesture translation.

### III. METHODS

This section presents our overall pipeline for the automated inference of surgical context and translation to gesture labels based on the video data as depicted in Figure 2. Surgical context can be inferred from the video or kinematic data by estimating the values of the state variables. In this work, we specifically focus on context inference solely based on video data as an independent method to verify gestures predicted from kinematic data or when kinematic data is not available. Our methods are presented for a case study of the JIGSAWS dataset [40] using the context labels from [25], but are applicable to other datasets and sets of gestures.

#### A. Tool and Object Segmentation

The detection of general and task-specific state variables for surgical context requires identifying the status and relative distance of the instruments and the objects of interest in a task. As shown in Figure 1b for the JIGSAWS tasks, these include the left and right graspers, needle, thread, and rings.

We modified the Deeplab V3 model [32] to perform binary segmentation that classifies the background vs. one object class in the video frames of a task trial. Specifically, we train separate binary classification models to classify background vs. left grasper, background vs. right grasper, background vs. needle, background vs. thread, and background vs. ring. The input to each model is a matrix  $A_{H \times W \times 3}$  representing an RGB image of a video frame with Height (H) and Width (W). The output is a binary matrix  $M_{H \times W}$  representing the segmentation mask with 0 for the background class, and 1 for the segmented object class. We need to infer the intersections between objects for generating context, which cannot be done with the existing multi-class segmentation models that classify each pixel to a single object class. Binary segmentation models for each object class enable the analysis of intersections and overlaps among separate object masks to infer interactions between objects.

For each object, we combine the data from all tasks to train a single model to classify that object in all tasks. We leveraged transfer learning by initializing the model with a ResNet-50 [41] backbone pre-trained on the COCO dataset [42]. We obtained tool and object annotations for

the JIGSAWS dataset and used a subset of 70 videos for fine-tuning the model. However, the test set for the whole pipeline was significantly limited since much of the data from JIGSAWS was needed to train the image segmentation models. We trained our models for up to 20 epochs using Adam optimization [43] with a learning rate of  $10^{-5}$ .

#### B. Automated Context Inference

The masks from the segmentation models provide us with information about the area and position of the instruments and objects which can enable state variable estimation at each frame. By calculating intersections and distances between the object masks in a given frame, we can detect interactions such as *contact* and *hold* as shown in Figure 1b.

In the mask matrices  $M_{H \times W}$  generated by the segmentation models, each element  $m_{hw} \in \{0, 1\}$  indicates if the pixel  $(h, w)$  belongs to an object mask. We first perform a pre-processing step on  $M$  to eliminate the noise around masks such as the needles and threads. Contour extraction is done to help eliminate the rough edges of the masks and improve intersection detection. This step uses the OpenCV library [44] to iteratively construct contours around every element  $m_{hw} \in M$ , thus reducing the input matrix to a list of points  $p \in C \subset M$  for each instrument class where  $C$  is the boundary of  $M$ . Using simplified polygons instead of binary masks greatly reduces the time needed to calculate intersections and distances between objects for each frame. We experimentally determined that dropping polygons with areas under 15 pixel units squared and smoothing the polygons using the Ramer–Douglas–Peucker (RDP) algorithm [45], [46] results in better accuracy based on training set.

Next, we detect overlaps between masks by taking a list of valid polygons and calculating a feature vector  $v$  of distances ( $D$ ) and intersection areas ( $Inter$ ) between pairs of input masks. The input polygons Left Grasper ( $LG$ ), Right Grasper ( $RG$ ), Thread ( $T$ ) are common for all tasks. Task-specific objects are the Needle ( $N$ ) appearing in Needle Passing and Suturing, the manually labeled Tissue Points ( $Ts$ ) representing the markings on the tissue where the needle makes contact in Suturing, and the Rings  $R$  in Needle Passing.

We define the distance functions  $D(I, J)$  and  $d(i, j)$  and the intersection function  $Inter(I, J)$  to, respectively, calculate the pixel distance between two object masks  $I$  and  $J$ , the pixel distance between the individual polygons  $i_1, j_1, \dots$  that constitute an object mask, and the area of intersection between two object masks  $I$  and  $J$ . For any object polygon  $I$  which is comprised of several polygon segments  $i_1, i_2, \dots, i_n$ , the distance to any other object  $J$  can be calculated as:  $D(I, J) = \text{average}([d(i, j) \text{ for } i \in I \text{ and } j \in J])$ . The intersection function  $Inter(I, J)$  is implemented using a geometric intersection algorithm from the Shapely [47] library. We also define the components  $I.x, I.y$  for an object  $I$  as the horizontal and vertical coordinates of the midpoint of its polygon  $I$ , calculated as the average of every point in  $I$ . In order to determine the Boolean function ( $\alpha$ ) for each grasper, if the distance between the manually labeled pixel coordinates of the grasper jaw ends was less than 18 pixels, then the grasper was closed ( $\neg\alpha$ ), else it was open ( $\alpha$ ).

$$\text{Left Hold} \begin{cases} 2 & \text{if } D(LG, N) < 1 \wedge \neg\alpha \\ 3 & \text{if } Inter(LG, T) > 0 \wedge \neg\alpha \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

$$\text{Left Contact} \begin{cases} 2 & \text{if } D(LG, N) < 1 \wedge \alpha \\ 3 & \text{if } Inter(LG, T) > 0 \wedge \alpha \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

$$\text{Needle} \begin{cases} 2 & \text{if } (Inter(Ts, N) > 0 \wedge N.x < Ts.x) \\ 1 & \text{if } (Inter(Ts, N) = 0 \vee N.x \geq Ts.x) \wedge \\ & (D(RG, T) > 1 \vee D(LG, N) > 1) \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The feature vector  $v = \langle D(LG, N), Inter(LG, T), \dots \rangle$  (see Figure 2) is then used to estimate the values of different state variables using a set of task-specific functions. An example set of functions is shown in Equations 1-3 for the state variables relating to the left robot arm and needle in Suturing task. A similar set of functions are used for the right arm. For example, if the distance between the left grasper and needle is less than one pixel ( $D(LG, N) < 1$ ) and the grasper is closed ( $\neg\alpha$ ), then a value of 2 is estimated for the *Left Hold* variable. Or the *Needle* state is detected as touching (2) when the relative horizontal distance of the needle polygon ( $N.x$ ) is less than the average (midpoint) of the tissue points ( $Ts.x$ ) and these two objects intersect ( $Inter > 0$ ).

The input sample rate of the context to gesture translation was 3Hz, so the final estimated variables were downsampled from 30Hz to 3Hz using a rolling mode for each state variable with a window of 10 frames.

### C. Context to Gesture Translation

The last step in our pipeline translates the automatically generated context labels into gesture labels. The input to the translation model is a 2-dimensional time series matrix  $\chi_{State \times n}$ , where *State* represents the 5 state variables describing the context (see Figure 1b) and  $n$  represents the total number of samples in the trial. We map each time step  $State_t$  to a corresponding gesture  $G_i$  in the JIGSAWS dataset. The translation output is a 1-dimensional time series  $Y_n \in \{G\}$

with each time step mapped to a gesture. We present two approaches based on domain knowledge and data.

1) *Finite State Machine Model*: Our first approach relies on a finite state machine (FSM) defined based on the knowledge of surgical tasks which directly relates context to gestures and is more explainable than deep learning models. The grammar graphs from [1] for each task were overlaid on top of the ideal context models from [25] so that each gesture could be mapped into the groups of contextual changes that happen as the result of executing the gesture (see Figure 3 for the Suturing task). For example, G2 (positioning needle) corresponds to a change from a ‘0’ to a ‘1’ in the fifth state variable. Or G4 (transferring needle from left to right) is the context sequence 20000  $\rightarrow$  20020  $\rightarrow$  20200  $\rightarrow$  02200  $\rightarrow$  00200 which means the needle is initially held in the left grasper, then touched and grasped by the right grasper, and released by the left grasper. In Figure 3, the G4 and G8 groupings overlap since G8 (orienting needle) is performed by passing the needle from the right to the left grasper and back to the right grasper while changing its orientation.

Given the context transcript of a trial, the FSM is evaluated for each context and a transition to the next gesture is detected if the input context is part of the next gesture. The FSM for each task was initialized in the ‘Start’ state since not all of the trials started with G1. Also, G11 was assumed to be last and so it was appended to the gestures following the last detected gesture. In addition, in the Suturing and Needle Passing tasks, G9 and G10 had low rates of occurrence and were not included in the final translation. This allowed us to focus only on state changes involving the needle and thus ignore grasps and touches of the thread and rings with the added benefit of simplifying the FSMs and limiting the total number of valid context changes.

We also consider gesture duration as a trigger for transitions between gestures. If the current gesture’s duration exceeds a certain threshold based on the average duration of that gesture class, a transition to the next gesture is enforced. This is to address the cases where a gesture transition does not happen due to inaccuracies in context detection. For example, the segmentation models tend to have lower accuracy in detecting the needle and thread states, leading to not detecting transitions that are dependent on those states.

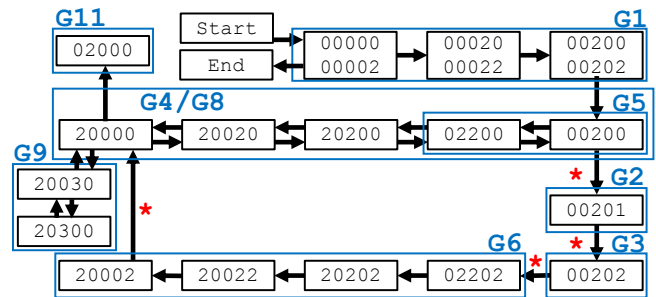


Fig. 3: Grouping and mapping of context to gestures in the grammar graph of the Suturing task. \* denotes transitions due to duration limits as follows:  $G2 > 6.0 \text{ s} \rightarrow G3$ ,  $G3 > 11.1 \text{ s} \rightarrow G6$ ,  $G4 > 5.2 \text{ s} \rightarrow G2$ ,  $G6 > 6.1 \text{ s} \rightarrow G4$ .

2) *LSTM Model*: Our second approach for translation of context to gesture transcripts relies on sequential deep learning methods to learn relationships in the data that are not captured by the FSM models. We trained an LSTM model to perform automated context to gesture translation for each task. We chose the LSTM model for its ability to learn temporal features. Specifically, we used a simple double layer LSTM network with 64 hidden units for the Suturing and Needle Passing tasks and 256 hidden units for the Knot Tying task. We used Adam optimization [43] and the cross entropy loss function to train the models. The hidden layers, number of hidden units and learning rates were determined by hyperparameter tuning. The final models were trained with the best model configurations and used to perform inference on the automatically generated context labels using the segmentation masks in the test set. Note that the LSTM model is a black box model and does not provide transparency like the FSM model in the previous section.

#### IV. EXPERIMENTAL EVALUATION

##### A. Experimental Setup

We use an 80/20 train/test split of the JIGSAWS dataset for evaluating our pipeline. The original videos are 30Hz and we obtained binary masks for the tools and objects at 2Hz which we then used to train/test the segmentation models. The LSTM networks are trained with the 3Hz context labels from [25]. We evaluate both the FSM and LSTM for context to gesture translation with the test set context labels.

The experiments were conducted on a PC with an Intel Core i7 CPU 3.60GHz, 32GB RAM, and an NVIDIA GeForce RTX 2080 Ti GPU running Ubuntu 18.04.2 LTS.

##### B. Metrics

The following metrics were used to evaluate the pipeline.

**Accuracy**: Accuracy is the ratio of samples with correct labels divided by the total number of samples in a trial.

**Edit Score**: Edit score is calculated using Equation 4 from [2] where the normalized Levenshtein edit distance,  $edit(G, P)$ , quantifies the number of insertions, deletions, and replacements needed to transform the sequence of predicted labels  $P$  to match the ground truth sequence of labels  $G$ . This is then normalized by the maximum length of the sequences so that a higher edit score is better.

$$\text{Edit Score} = \left(1 - \frac{edit(G, P)}{\max(\text{len}(G), \text{len}(P))}\right) \times 100 \quad (4)$$

**Intersection over Union (IOU)**: Mean IOU, as calculated in Equation 5, is the standard for assessing the segmentation and translation models [42].

$$IOU = TP / (TP + FP + FN) \quad (5)$$

Each predicted segment is matched to a corresponding segment in the ground truth. Then, the average IOU for each class is calculated and the mean of class IOUs is returned.

TABLE I: Tool and object segmentation performance on the test set (mean IOU for each object class) on the MICCAI18 (M) and JIGSAWS Suturing (S), Needle Passing (NP), and Knot Tying (KT) tasks.

Model	Data	Graspers		Objects		
		Left	Right	Needle	Thread	Ring
Deeplab V3+ [27] U-net [27]	M [27]	0.78		0.014	<b>0.48</b>	N/A
		0.72		0.02	0.33	N/A
Mobile-U-Net [37]	S	<b>0.82</b>		<b>0.56</b>	N/A	N/A
Trained UNet [38]	S	0.69		N/A	N/A	N/A
	NP	0.66		N/A	N/A	N/A
Trained LinkNet34	KT	0.80		N/A	N/A	N/A
Deeplab V3 (ours)	S	0.71	<b>0.64</b>	<b>0.19</b>	<b>0.52</b>	N/A
	NP	0.61	0.49	0.09	0.25	<b>0.37</b>
	KT	<b>0.74</b>	0.61	N/A	0.44	N/A

##### C. Results

1) *Tool and Object Segmentation*: Table I shows the performance of our segmentation models in comparison to the related work. Although the MICCAI 18 challenge [27] dataset is from real porcine procedures, and differs from the JIGSAWS dataset collected from dry-lab experiments, it has similar objects including the grasper (similar to the graspers in JIGSAWS), needle and thread. The Deeplab V3+ model achieved the best performance on the thread class. The top models from MICCAI 18 do not perform as well as our binary models on the needle and thread classes in the Suturing task. However, the Mobile-U-Net [37] achieved the highest performance for grasper and needle segmentation in the JIGSAWS Suturing task. [38] reported tool segmentation IOUs for all the JIGSAWS tasks with up to 0.8 for KT using a Trained LinkNet34, but did not do object segmentation. Among the JIGSAWS tasks, we achieved the best performance in Suturing for the right grasper, needle and thread, while the model performance on the Needle Passing task was the worst. This is likely due to Needle Passing’s background having less contrast with the foreground compared to the other two tasks as shown in Figure 2). We can also see that the needle and thread masks are thinner compared to the grasper masks. So, the mask boundary errors could contribute to a lower score for the needle and thread classes. The estimated time for segmenting the whole JIGSAWS dataset is 8.6 hours.

2) *Automated Context Inference*: Table II shows the performance of the context inference method in terms of IOU achieved for each state variable with the predicted segmentation masks and the ground truth masks from crowd-sourcing.

The left column of Table II shows that left and right contact have higher IOUs compared to left and right hold, and the needle or knot state has the lowest IOU. This is because errors in estimating the position of the grasper jaw ends affect accurate inference of the hold state, while contact is relatively simple by finding if the two masks intersect. Better performance in detecting contact compared to hold states is also observed in the right column of Table II, where ground truth segmentation masks are used. Hence, the lower performance of the left hold and right hold could primarily be due to the difficulty in detecting these states.

TABLE II: State variable IOU with consensus context using predicted masks from DeepLab V3 and ground truth masks

	Predicted Masks					Avg	Ground Truth Masks					Avg
	Left Hold	Left Contact	Right Hold	Right Contact	Needle or Knot		Left Hold	Left Contact	Right Hold	Right Contact	Needle or Knot	
Suturing	0.48	0.75	<b>0.60</b>	0.87	0.30	0.60	0.52	0.77	<b>0.61</b>	0.87	0.39	0.63
Needle Passing	0.40	<b>0.97</b>	0.18	<b>0.95</b>	<b>0.39</b>	0.58	0.42	<b>0.97</b>	0.19	<b>0.94</b>	<b>0.41</b>	0.59
Knot Tying	<b>0.75</b>	0.72	0.57	0.78	<b>0.59</b>	<b>0.68</b>	<b>0.83</b>	0.77	<b>0.61</b>	0.79	<b>0.62</b>	<b>0.72</b>
Avg	0.54	<b>0.81</b>	0.45	<b>0.87</b>	0.43		0.59	<b>0.84</b>	0.47	<b>0.87</b>	0.47	

TABLE III: Gestures translated from the automatic context inference given masks from the Deeplab V3 models

Task	Model	Gestures from Predicted Context Labels			Gestures from Consensus Context Labels		
		Accuracy (%)	Edit Score	IOU	Accuracy (%)	Edit Score	IOU
Suturing	FSM	<b>40.8</b>	<b>67.1</b>	<b>0.28</b>	<b>66.3</b>	<b>84.4</b>	<b>0.48</b>
	LSTM	25.4	24.9	0.17	38.8	34.7	0.26
Suturing (vid)	Zero-shot [10]	<b>56.6</b>	<b>61.7</b>				
Suturing (kin)	TSSC-DL [11]	49.7	32.8				
Needle Passing	FSM	<b>18.0</b>	<b>76.2</b>	<b>0.12</b>	<b>70.1</b>	<b>88.7</b>	<b>0.54</b>
	LSTM	14.8	20.1	0.02	17.0	20.0	0.04
Knot Tying	FSM	<b>42.9</b>	<b>70.7</b>	<b>0.43</b>	<b>54.6</b>	<b>91.5</b>	<b>0.43</b>
	LSTM	36.5	23.8	0.17	50.8	49.2	0.28

For the needle/knot state, we need to detect if the needle is in the fabric/tissue for the Suturing task, in/out of the ring for the Needle Passing task, and if the knot is loose or tight in the Knot Tying task. Detecting the state of the needle and knot is difficult even with the ground truth segmentation masks in the right column of Table II. This is because the needle and thread have the lowest segmentation performance compared to graspers as shown in Table I. The total time to perform automatic context inference is estimated to be about 30 seconds for the whole JIGSAWS dataset.

3) *Context to Gesture Translation*: The right column of Table III shows the performance of the FSM and LSTM methods in translating ground truth context labels to gestures. The FSM model achieves higher accuracies and edit scores than the LSTM. The left column of Table III shows the performance of the overall pipeline with automated context labels. We see that using automated context from predicted masks degrades the performance of both models because the segmentation models perform poorly at generating masks for the needle and for all tools and objects in Needle Passing. This effect is propagated through the pipeline, resulting in low accuracies and IOUs. The FSM generally outperforms the LSTM likely due to its knowledge-based structure and setting limits on gesture durations that prevent the model from becoming stuck in any one gesture even with degraded context labels. The FSM pipeline achieves accuracies lower than unsupervised models from [10] and [11] for Suturing, but outperforms them in terms of edit score. These observations suggest that there are benefits to incorporating knowledge into context to gesture translation that can make the model more robust to degraded context labels. However, the FSM is manually developed based on domain knowledge and relies on defined inputs and transitions while the LSTM requires labeled data for training. The time to generate the entire JIGSAWS gesture translation from context is less than 3 minutes for both models.

## V. DISCUSSION AND CONCLUSIONS

Our proposed pipeline for automated inference of surgical context and translation to gesture labels can perform automatic and explainable gesture inference given video segmentation masks. It can be used as an efficient and fast inference method by significantly shortening manual gesture labeling time (~9 hours vs. ~26 hours for the case study of the JIGSAWS dataset). We rely on models pre-trained on general images and publicly-available datasets which lowers the cost of manually labeling video data and makes our model generalizable to other datasets and tasks.

For the case study of JIGSAWS, our binary segmentation models achieve comparable performance to state-of-the-art models on the grasper and thread classes, and better performance on the needle class. However, they do not perform well enough for the needle and thread classes which are important for accurate context inference. Our context inference method also does not perform equally well for all the states. Given the ground truth segmentation masks, it achieves ~85% IOU for states such as left/right contact, but only ~45% IOU for the needle/knot state. The FSM and LSTM models for context to gesture translation have better performance given ground truth context labels compared to predicted context which may be due to imperfect models at each stage of the pipeline and error propagation.

Manual annotations for the grasper end points and tissue points were used for context inference. Also, our method relies on 2D images to infer context from a 3D environment which can particularly complicate detecting the contact states. Future work will focus on addressing these limitations and improving the performance and robustness of the overall pipeline to apply it to runtime error detection [16], [17].

## ACKNOWLEDGMENT

This work was supported in part by the National Science Foundation grants DGE-1829004 and CNS-2146295.

## REFERENCES

- [1] N. Ahmidi, L. Tao, S. Sefati, Y. Gao, C. Lea, B. B. Haro, L. Zappella, S. Khudanpur, R. Vidal, and G. D. Hager, "A dataset and benchmarks for segmentation and recognition of gestures in robotic surgery," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 9, pp. 2025–2041, 2017.
- [2] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in *Computer Vision—ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8–10 and 15–16, 2016, Proceedings, Part III 14*. Springer, 2016, pp. 47–54.
- [3] C. Lea, A. Reiter, R. Vidal, and G. D. Hager, "Segmental spatiotemporal cnns for fine-grained action segmentation," pp. 36–52, 2016.
- [4] R. DiPietro, C. Lea, A. Malpani, N. Ahmidi, S. S. Vedula, G. I. Lee, M. R. Lee, and G. D. Hager, "Recognizing surgical activities with recurrent neural networks," pp. 551–558, 2016.
- [5] R. DiPietro, N. Ahmidi, A. Malpani, M. Waldram, G. I. Lee, M. R. Lee, S. S. Vedula, and G. D. Hager, "Segmenting and classifying activities in robot-assisted surgery with recurrent neural networks," *International journal of computer assisted radiology and surgery*, vol. 14, no. 11, pp. 2005–2020, 2019.
- [6] I. Funke, S. Bodenstedt, F. Oehme, F. von Bechtolsheim, J. Weitz, and S. Speidel, "Using 3d convolutional neural networks to learn spatiotemporal features for automatic surgical gesture recognition in video," in *Medical Image Computing and Computer Assisted Intervention—MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part V 22*. Springer, 2019, pp. 467–475.
- [7] F. Despinoy, D. Bouget, G. Forestier, C. Penet, N. Zemiti, P. Poignet, and P. Jannin, "Unsupervised trajectory segmentation for surgical gesture recognition in robotic training," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1280–1291, 2015.
- [8] M. J. Fard, S. Ameri, R. B. Chinnam, and R. D. Ellis, "Soft boundary approach for unsupervised gesture segmentation in robotic-assisted surgery," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 171–178, 2016.
- [9] S. Krishnan, A. Garg, S. Patil, C. Lea, G. Hager, P. Abbeel, and K. Goldberg, "Transition state clustering: Unsupervised surgical trajectory segmentation for robot learning," *The International Journal of Robotics Research*, vol. 36, no. 13–14, pp. 1595–1618, 2017.
- [10] J. D. Jones, T. S. Kim, M. Peven, J. Bai, Z. Xiao, Y. Zhang, W. Qiu, A. Yuille, and G. D. Hager, "Zero-shot recognition of complex action sequences," *ADVANCES IN WATER RESOURCES, SOUTHAMP-TON, UK, COMPUTATIONAL MECHANICS PUBLICATIONS*, 2019.
- [11] L. Clopton, E. Mavroudi, M. Tsakiris, H. Ali, and R. Vidal, "Temporal subspace clustering for unsupervised action segmentation," *CSMR REU*, pp. 1–7, 2017.
- [12] Y. Gao, S. S. Vedula, C. E. Reiley, N. Ahmidi, B. Varadarajan, H. C. Lin, L. Tao, L. Zappella, B. Béjar, D. D. Yuh *et al.*, "Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling," in *MICCAI Workshop: M2CAI*, vol. 3, 2014, p. 3.
- [13] B. van Amsterdam, M. Clarkson, and D. Stoyanov, "Gesture recognition in robotic surgery: a review," *IEEE Transactions on Biomedical Engineering*, 2021.
- [14] L. Tao, E. Elhamifar, S. Khudanpur, G. D. Hager, and R. Vidal, "Sparse hidden markov models for surgical gesture classification and skill evaluation," in *International conference on information processing in computer-assisted interventions*. Springer, 2012, pp. 167–177.
- [15] B. Varadarajan, C. Reiley, H. Lin, S. Khudanpur, and G. Hager, "Data-derived models for segmentation with application to surgical assessment and training," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2009, pp. 426–434.
- [16] M. S. Yasar and H. Alemzadeh, "Real-time context-aware detection of unsafe events in robot-assisted surgery," pp. 385–397, 2020.
- [17] Z. Li, K. Hutchinson, and H. Alemzadeh, "Runtime detection of executional errors in robot-assisted surgery," p. 3850–3856, 2022. [Online]. Available: <https://doi.org/10.1109/ICRA46639.2022.9812034>
- [18] M. Ginesi, N. Sansonetto, and P. Fiorini, "Overcoming some drawbacks of dynamic movement primitives," *Robotics and Autonomous Systems*, vol. 144, p. 103844, 2021.
- [19] D. Kitaguchi, N. Takeshita, H. Hasegawa, and M. Ito, "Artificial intelligence-based computer vision in surgery: Recent advances and future perspectives," *Annals of Gastroenterological Surgery*, vol. 6, 10 2021.
- [20] K. Hutchinson, Z. Li, L. A. Cantrell, N. S. Schenkman, and H. Alemzadeh, "Analysis of executional and procedural errors in dry-lab robotic surgery experiments," *arXiv preprint arXiv:2106.11962*, 2021.
- [21] B. van Amsterdam, M. J. Clarkson, and D. Stoyanov, "Multi-task recurrent neural network for surgical gesture recognition and progress prediction," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 1380–1386.
- [22] D. Meli and P. Fiorini, "Unsupervised identification of surgical robotic actions from small non homogeneous datasets," *arXiv preprint arXiv:2105.08488*, 2021.
- [23] L. Li, X. Li, S. Ding, Z. Fang, M. Xu, H. Ren, and S. Yang, "Sirnet: Fine-grained surgical interaction recognition," *IEEE Robotics and Automation Letters*, 2022.
- [24] C. I. Nwoye, T. Yu, C. Gonzalez, B. Seeliger, P. Mascagni, D. Mutter, J. Marescaux, and N. Padoy, "Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos," *Medical Image Analysis*, vol. 78, p. 102433, 2022.
- [25] K. Hutchinson, I. Reyes, Z. Li, and H. Alemzadeh, "COMPASS: a formal framework and aggregate dataset for generalized surgical procedure modeling," *arXiv preprint arXiv:2209.06424*, 2022.
- [26] M. Allan, A. Shvets, T. Kurmann, Z. Zhang, R. Duggal, Y.-H. Su, N. Rieke, I. Laina, N. Kalavakonda, S. Bodenstedt *et al.*, "2017 robotic instrument segmentation challenge," *arXiv preprint arXiv:1902.06426*, 2019.
- [27] M. Allan, S. Kondo, S. Bodenstedt, S. Leger, R. Kadkhodamohammadi, I. Luengo, F. Fuentes, E. Flouty, A. Mohammed, M. Pedersen *et al.*, "2018 robotic scene segmentation challenge," *arXiv preprint arXiv:2001.11190*, 2020.
- [28] D. Neumuth, F. Loebe, H. Herre, and T. Neumuth, "Modeling surgical processes: A four-level translational approach," *Artificial intelligence in medicine*, vol. 51, no. 3, pp. 147–161, 2011.
- [29] T. Neumuth, G. Strauß, J. Meixensberger, H. U. Lemke, and O. Burgert, "Acquisition of process descriptions from surgical interventions," in *Database and Expert Systems Applications: 17th International Conference, DEXA 2006, Kraków, Poland, September 4–8, 2006. Proceedings 17*. Springer, 2006, pp. 602–611.
- [30] M. S. Yasar, D. Evans, and H. Alemzadeh, "Context-aware monitoring in robotic surgery," in *2019 International Symposium on Medical Robotics (ISMR)*. IEEE, 2019, pp. 1–7.
- [31] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 801–818.
- [32] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
- [33] Z.-L. Ni, G.-B. Bian, X.-H. Zhou, Z.-G. Hou, X.-L. Xie, C. Wang, Y.-J. Zhou, R.-Q. Li, and Z. Li, "Raunet: Residual attention u-net for semantic segmentation of cataract surgical instruments," in *International Conference on Neural Information Processing*. Springer, 2019, pp. 139–149.
- [34] 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: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III 18*. Springer, 2015, pp. 234–241.
- [35] V. Iglovikov and A. Shvets, "Ternausnet: U-net with vgg11 encoder pre-trained on imagenet for image segmentation," *arXiv preprint arXiv:1801.05746*, 2018.
- [36] A. Chaurasia and E. Culurciello, "Linknet: Exploiting encoder representations for efficient semantic segmentation," in *2017 IEEE visual communications and image processing (VCIP)*. IEEE, 2017, pp. 1–4.
- [37] J. K. H. Andersen, K. L. Schwaner, and T. R. Savarimuthu, "Real-time segmentation of surgical tools and needle using a mobile-u-net," in *2021 20th International Conference on Advanced Robotics (ICAR)*. IEEE, 2021, pp. 148–154.
- [38] D. Papp, R. N. Elek, and T. Haidegger, "Surgical tool segmentation on the jigsaws dataset for autonomous image-based skill assessment," in

2022 IEEE 10th Jubilee International Conference on Computational Cybernetics and Cyber-Medical Systems (ICCC). IEEE, 2022, pp. 000 049–000 056.

- [39] A. Jin, S. Yeung, J. Jopling, J. Krause, D. Azagury, A. Milstein, and L. Fei-Fei, “Tool detection and operative skill assessment in surgical videos using region-based convolutional neural networks,” pp. 691–699, 2018.
- [40] Computational Interaction and Robotics Laboratory, “JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS).”
- [41] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [42] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *European conference on computer vision*. Springer, 2014, pp. 740–755.
- [43] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [44] G. Bradski, “The OpenCV Library,” *Dr. Dobb's Journal of Software Tools*, 2000.
- [45] U. Ramer, “An iterative procedure for the polygonal approximation of plane curves,” *Computer Graphics and Image Processing*, vol. 1, no. 3, pp. 244–256, 1972. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0146664X72800170>
- [46] D. H. Douglas and T. K. Peucker, “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,” *Cartographica: the international journal for geographic information and geovisualization*, vol. 10, no. 2, pp. 112–122, 1973.
- [47] S. Gillies *et al.*, “Shapely: manipulation and analysis of geometric objects,” 2007–. [Online]. Available: <https://github.com/shapely/shapely>