Evaluating the Task Generalization of Temporal Convolutional Networks for Surgical Gesture and Motion Recognition using Kinematic Data

Kay Hutchinson[1], Ian Reyes[2], Zongyu Li[1], and Homa Alemzadeh[1]

***Abstract*—Fine-grained activity recognition enables explainable analysis of procedures for skill assessment, autonomy, and error detection in robot-assisted surgery. However, existing recognition models suffer from the limited availability of annotated datasets with both kinematic and video data and an inability to generalize to unseen subjects and tasks. Kinematic data from the surgical robot is particularly critical for safety monitoring and autonomy, as it is unaffected by common camera issues such as occlusions and lens contamination. We leverage an aggregated dataset of six dry-lab surgical tasks from a total of 28 subjects to train activity recognition models at the gesture and motion primitive (MP) levels and for separate robotic arms using only kinematic data. The models are evaluated using the LOUO (Leave-One-User-Out) and our proposed LOTO (Leave-One-Task-Out) cross validation methods to assess their ability to generalize to unseen users and tasks respectively. Gesture recognition models achieve higher accuracies and edit scores than MP recognition models. But, using MPs enables the training of models that can generalize better to unseen tasks. Also, higher MP recognition accuracy can be achieved by training separate models for the left and right robot arms. For task-generalization, MP recognition models perform best if trained on similar tasks and/or tasks from the same dataset.**

***Index Terms*—Medical Robots and Systems, Recognition, Kinematics**

I. Introduction

In robot-assisted surgery (RAS), modeling and analysis at the gesture and action levels of the surgical hierarchy [1], [2] is performed to gain a better understanding of surgical activity and improve skill assessment [3], [4], error detection [5]–[8], and autonomy [9]. Towards these applications, automated segmentation and classification of surgical workflow has been an active area of research [10]. [11] and [12] provide comprehensive summaries of the recent works at the gesture and action levels. However, previous works and comparisons among them have been restricted by differing gesture definitions [11] and limited diversity in the numbers of subjects, trials, and *tasks* across the existing datasets.

Recent works in gesture recognition have each defined their own sets of gestures for their own datasets [13]–[17] with limited overlap between gestures. On the other hand, works on recognition of fine-grained surgical actions focus on action triplets (verb, instrument, tissue/object) [18]–[20], representing surgical instrument and tissue interactions in endoscopic videos. While gesture recognition has been done with kinematic and/or video data [11], recent work on action triplet recognition has mainly focused on video data of surgical procedures [19], [20]. To leverage finer-grained action recognition in safety monitoring and autonomy applications, in this paper we examine verb-only predictions based on kinematic data. Kinematic data is particularly important for safety analysis [7], [8], error detection [6], [11], and improved recognition accuracy using multi-modal data [21], [22], since it is unaffected by common camera issues such as occlusions, lens contamination, and smoke [5], [23], [24]. Plus, using fewer data types can reduce computational cost and enable real-time applications [25].

To address the challenge of limited datasets, Hutchinson et al. presented a new dataset, called COMPASS [12], which aggregates six dry-lab surgical training tasks from the JIGSAWS [26], DESK [16], and ROSMA [27] datasets by providing standardized *context* and *motion primitive (MP)* labels for all the tasks. MPs are a standardized set of actions (e.g., push) whose execution results in changes of surgical context, which is comprised of important state variables describing physical status and interactions of tools and tissues/objects (e.g., needle in tissue). Some of the tasks in the dataset share similar objects and goals enabling their aggregation and comparison. The standardized labels in COMPASS can support aggregated analysis of datasets and combining data from contextually similar tasks for improved activity recognition and error detection [7], [8], [11].

In this paper, we use the COMPASS dataset to study the effect of label granularity on activity recognition performance and generalization across users, tasks, and datasets for RAS with a case study of Temporal Convolutional Networks (TCN) [28]. Specifically, we make the following contributions:

- We compare the performance of existing activity recognition models in a case study of TCN using only kinematic data at different levels of the surgical hierarchy, specifically, the gesture and motion primitive levels, and for

Manuscript received: February 2, 2023; Revised May 27, 2023; Accepted June 16, 2023.

This paper was recommended for publication by Pietro Valdastri upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in part by the National Science Foundation grants DGE-1842490, DGE-1829004, and CNS-2146295 and by the Engineering-in-Medicine center at the University of Virginia.

[1]Kay Hutchinson, Zongyu Li, and Homa Alemzadeh are with the Department of Electrical and Computer Engineering, University of Virginia, Charlottesville, VA 22903 USA {kch4fk, zl7qw, ha4d}@virginia.edu

[2]Ian Reyes was with the Department of Computer Science, University of Virginia, Charlottesville, VA 22903 USA. He is now with IBM. ir6mp@virginia.edu

Digital Object Identifier (DOI): see top of this page.

separate left and right sides of the robot vs. both sides combined.
- We introduce the Leave-One-Task-Out (LOTO) cross validation method to measure the ability of surgical activity recognition models to generalize to an unseen task, since current datasets do not include all of the surgical tasks that a model may see when it is deployed.
- We perform the first evaluation of a surgical activity recognition model trained on multiple tasks with data combined from different datasets by comparing model performance using the existing LOUO method as well as our proposed LOTO cross validation method.

The insights from our analysis can guide the development of future surgical activity recognition and error detection models. The aggregated dataset and code to train and evaluate the recognition models are publicly available at https://github.com/UVA-DSA/COMPASS.

II. Background

A. Levels of Granularity in Surgical Procedures

Surgical process modeling [1], [2] decomposes surgical procedures into smaller units such as steps, tasks, gestures, and actions as shown in Figure 1. We refer to units at any level of the surgical hierarchy as "activities". Gestures are defined as "intentional surgical activit[ies] resulting in a perceivable and meaningful outcome" (e.g., pushing needle through tissue) [26] and usually include the semantics of both the activity and the underlying physical context in their definition. We also consider surgical actions (i.e., the verbs of action triplets [20], [29]) which are atomic units of activity or lower level motions (e.g., grasp, push) based on kinematic data, but without the semantics of physical context or the types and status of interacting tools and objects/tissues (e.g., needle through tissue) based on video data [1].

Existing activity recognition models have been mostly task-specific and restricted to specific datasets and gesture definitions. For example, the majority of previous works have used the JIGSAWS dataset and gesture definitions [26]. To address this, [12] defined a finer-grained set of motion primitives (MPs) as generalizable surgical actions to enable comparative analysis between tasks and datasets. MPs are similar in granularity and definition to the action triplets defined by [20].

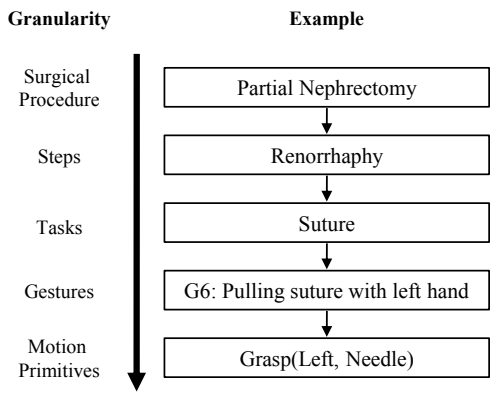


Fig. 1: Surgical Hierarchy. Adapted from [7]

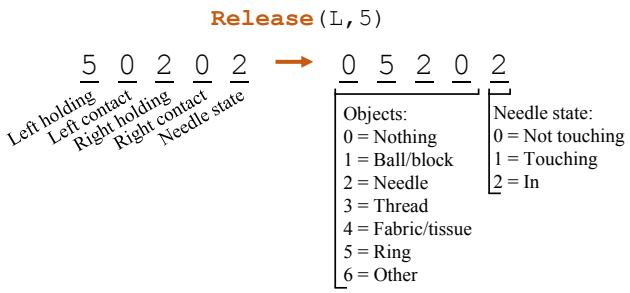
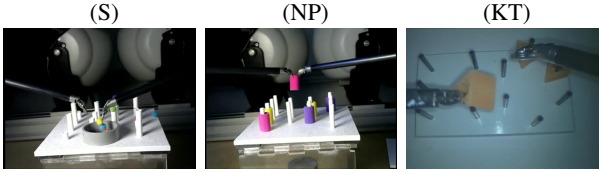


Fig. 2: Context states and object encodings for a "Release" motion primitive from the Needle Passing task [12].

(a) Suturing (S) (b) Needle Passing (NP) (c) Knot Tying (KT)



(d) Pea on a Peg (PoaP) (e) Post and Sleeve (PaS) (f) Peg Transfer (PT)

Fig. 3: COMPASS tasks: S, NP, and KT from JIGSAWS [26]; PoaP and PaS from ROSMA [27]; PT from DESK [16].

Each MP consists of a verb (e.g., Grasp), the tool that is used (e.g., left grasper), and the object with which the tool interacts (e.g., needle). The left and right graspers are abbreviated as 'L' and 'R', and the object encodings are shown in Figure 2 for an example MP and physical context. Table V shows the set of MPs and the number of samples in each MP class and task.

B. COMPASS Dataset

We use the COMPASS dataset [12] since it has different dry-lab tasks from multiple datasets and kinematic data from da Vinci surgical robots with which to train our surgical activity recognition models. We compare the performance of these models at the gesture and MP granularities. The COMPASS dataset contains kinematic and video data at 30 Hz for a total of six tasks from three different datasets as described in Table I. The tasks are: Suturing (S), Needle Passing (NP), Knot Tying (KT), Peg Transfer (PT), Post and Sleeve (PaS), and Pea on a Peg (PoaP) as shown in Figure 3. Context and MP labels are present for all trials, but gesture labels are only available for trials in the JIGSAWS and DESK datasets. To generate separate left and right label sets, MPs performed by each arm of the robot are split into new transcripts. Also, an 'Idle' MP is defined and used to fill the gaps created by the separation so that every kinematic sample has a label. An example segment of a Needle Passing trial with each label type is shown in Figure 4. This also shows the discrepancy in the G3 boundary noted by [7] where the Push(Needle, Ring) MP is in G2 rather than G3.

TABLE I: Number of subjects and trials and types of annotations for each task in the COMPASS dataset: Suturing (S), Needle Passing (NP), Knot Tying (KT), Peg Transfer (PT), Post and Sleeve (PaS), and Pea on a Peg (PoaP).

Dataset	JIGSAWS [26]			DESK [30]	ROSMA [27]	
Tasks	S	NP	KT	PT	PaS	PoaP
Trials	39	28	36	47	65	71
Subjects	8			8	12	
Gesture Labels	✓			✓		
MP Labels	✓			✓	✓	

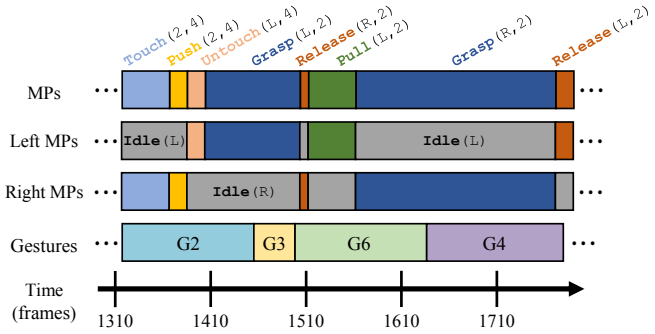


Fig. 4: Example of alignment between MPs and gestures in a Needle Passing trial that also shows the G3 boundary discrepancy noted by [7] where the 'Push' MP is not part of G3. From [26], G2: positioning needle, G3: pushing needle through tissue, G4: transferring needle from left to right, G6: pulling suture with left hand. Figure best viewed in color.

III. Related Work

Surgical workflow segmentation has been examined in different datasets with different tasks and at different levels of granularity as summarized in Table II.

Datasets and Tasks: Recent works in surgical activity recognition perform comparative evaluation of their models across different datasets. For example, [10] developed the TAPIR model and found that it performed better on the MISAW dataset [31] than their PSI-AVA dataset for phase and step recognition, but did not examine the reason for this. [6] evaluated an LSTM using LOSO cross validation on the JIGSAWS dataset and their own dataset of Block Transfer on the RAVEN II. The LSTM achieved a higher accuracy for the Block Transfer task since it was a simpler task with a larger amount of data compared to the JIGSAWS tasks. [8] found that combining data from the Suturing and Needle Passing tasks in JIGSAWS could improve error detection performance because the gestures were kinematically similar. [38] found that gesture recognition models trained on the JIGSAWS dataset did not generalize well to other dry-lab or clinical data. Whereas previous works did not combine data from multiple datasets or tasks since the label definitions differed, in this paper we examine such aggregation in training surgical activity recognition models.

Label Granularities: Surgical workflow recognition has been examined at different levels of granularity as listed in the fifth column of Table II. Note that there are inconsistencies in label and granularity definitions across datasets. For example, the *tasks* of Suturing, Knot Tying, and Peg Transfer in JIGSAWS and DESK are considered *phases* in MISAW [31] and PETRAW [36]. [13] trained a GRU for gesture and maneuver recognition on the JIGSAWS and MISTIC-SL datasets, respectively. Although the different datasets had different labels, the lower-level gesture recognition model had a higher error rate. The MISAW challenge [31] and HeiChole benchmark [33] datasets were labeled at multiple levels as well as the PSI-AVA dataset [10]. The best performing models from these works all showed decreasing performance metrics for finer-grained labels which highlights a significant challenge for fine-grained recognition. Interestingly, [31] found that multi-granularity recognition models performed better because such models may be learning that certain activities only occur during specific phases and steps. Also, recent works on action triplet recognition in laparoscopic procedures focus on concurrent phase, step, and action recognition [36]. The poor performance of activity recognition models is a barrier to clinical applications, but understanding the relationship between granularity levels can address this challenge and guide model development. This work closes a gap between the gesture and action levels of the hierarchy by evaluating and comparing the performance of an activity recognition model at those granularities.

IV. Methods

This section presents our methods for the construction and evaluation of gesture and MP recognition models.

A. Data Pre-processing

The input to the activity recognition model is the time-series kinematic data, x_t, and the output is a transcript of class labels, y_t, one for each time-series sample, where each class label is selected from the finite set of gestures or MPs. We experimented with different combinations of kinematic variables as inputs to the activity recognition models (while hyperparameter and cross validation settings were kept constant) and found that using only the position, linear velocity, and gripper angle kinematic variables resulted in the best performance. This is consistent with the best performing gesture recognition models that relied on kinematic data as reported in [11]. The stride was 1, so there was no downsampling, and the kinematic data and gesture and MP labels were all at 30 Hz.

B. Surgical Activity Recognition Model

One of the fastest and best performing models that used only kinematic data for gesture recognition in [11] was the Temporal Convolutional Network (TCN). The TCN is also used as a component in more complex state-of-the-art models such as MA-TCN [22] and MRG-Net [32]. Thus, as a case study, we adopt the TCN model from [28] for activity recognition at both gesture and MP levels. This model has an encoder-decoder structure, each consisting of three convolutional layers with pooling, channel normalization, and upsampling. As in [28],

TABLE II: Surgical workflow segmentation models that considered multiple datasets and label granularities.

Paper	Dataset	Data Type	Tasks	Label Levels	Best Teams/Models and Performance	
MISAW Challenge 2021 [31]	MISAW	Kinematics and/or Video	Anastomosis	Phases	MedAIR [32]	AD-Accuracy: 96.5%
				Steps	MedAIR [32]	AD-Accuracy: 84.0%
				Activities	NUSControl Lab and UniandesBCV	AD-Accuracy: ∼64%
				Multigranularity	NUSControl Lab	AD-Accuracy: ∼72%
HeiChole benchmark 2021 [33]	EndoVis 2019	Video	Laparoscopic cholecystectomy	Phase	HIKVision and CUHK	F1 Score: ∼65%
				Actions	Wintegral	F1 Score: 23.3%
Valderrama 2022 [10]	PSI-AVA	Video	Radical prostatectomy	Phase	TAPIR	mAP: 56.6%
				Step		mAP: 45.6%
				Action		mAP: 23.6%
DiPietro 2019 [13]	JIGSAWS	Kinematics	Suturing	Gestures	GRU	Error rate: 15.2% Edit distance: 8.4
	MISTIC-SL		Knot Tying	Maneuvers		Error rate: 8.6% Edit distance: 9.3
Multi-modal attention [22]	JIGSAWS	Kin+Vid	Suturing	Gestures	MA-TCN (Acausal)	Accuracy: 86.8% Edit: 91.4
	own (dV)					Accuracy: 80.9% Edit: 79.6
Gesture Recognition Survey [11]	JIGSAWS	Kinematics	Suturing	Gestures	MS-RNN [34]	Acc: 90.2% Edit Score*: 89.5
		Video			Symm dilation+attention [35]	Acc: 90.1% Edit Score: 89.9%
		Kin+Vid			Fusion-KV [21]	Acc: 86.3% Edit Score: 87.2
PETRAW Challenge 2021 [36]	PETRAW	Video	Peg Transfer	Phases, Steps, and Activities	SK	AD-Accuracy: 90.8%
		Kinematics			MedAIR	AD-Accuracy: 90.7%
		Segmentation			SK	AD-Accuracy: 88.5%
		Vid+Kin			NCC NEXT	AD-Accuracy: 93.1%
		Vid+Kin+Seg			NCC NEXT	AD-Accuracy: 93.1%
Sim2Real Gesture Classification [16], [30]	DESK	Kinematics	Peg Transfer	Gestures	RF	Simulator Acc: 86%
						Robot Acc: 95%
						Sim2Real (0% Real) Acc: 34%
						Sim2Real (18% Real) Acc: 85%
CholecTriplet2021 Challenge [37]	CholecT50	Video	Laparoscopic cholecystectomy	Action Triplets	Trequartista	AP_V: 52.9 AP_{IVT}: 38.1

* Normalized by maximum number of segments in any ground-truth sequence.

the kernel size is set to the average duration of the shortest activity class (e.g., gesture or MP), and the three layers have 32, 64, and 96 filters respectively. We used the cross-entropy loss function and Adam optimizer [39].

The learning rate and weight decay hyperparameters for all TCN models were selected based on a grid search of values by training on the JIGSAWS dataset with gesture labels for each cross validation setup. For LOUO models, the learning rate was 0.00005 and the weight decay was 0.0005. For LOTO models, the learning rate was 0.0001 and the weight decay was 0.001. These values were fixed for all models of their respective cross validation setup to analyze the effect of different training and label sets on model performance.

We compare the performance of the TCN when trained with four different sets of labels: gestures, MPs for only the left side (Left MPs), MPs for only the right side (Right MPs), and MPs for both sides together (MPs).

C. Model Generalization

We evaluate the generalization of the recognition models to unseen users/subjects and surgical tasks using two cross validation setups: Leave-One-User-Out (LOUO) from [40] and our novel Leave-One-Task-Out (LOTO).

1) Leave-One-User-Out (LOUO): LOUO is the standard cross validation setup for comparing gesture recognition models and is preferred over the Leave-One-Supertrial-Out (LOSO) method as it measures a model's ability to generalize to an unseen user as expected of a deployed model [11]. Since tasks from different aggregated datasets in COMPASS do not share the same subjects, we extended the LOUO setup from JIGSAWS [40] to include the new subjects, resulting in a maximum of 28 folds (corresponding to 28 users) when the model was trained on data from all tasks.

2) Leave-One-Task-Out (LOTO): Existing datasets represent a limited number of trials, subjects, and tasks. This means that machine learning models trained on them will see subjects, trials, and *tasks* that could be very different when they are deployed. In order to assess a model's ability to generalize to an unseen task, we introduce the Leave-One-Task-Out (LOTO) cross validation method.

In the LOTO setup, all of the data for one *task* was held out as the test set while the model was trained on all of the data for a set of other tasks. Thus the model would be tested on all the trials of all subjects from an unseen task. For an example fold, a model could be trained on NP, KT, PT, PaS, and PoaP and tested on S. This differs from the LOSO setup where a model would be tested on unseen *trials* from a known subject

of a known task. Similar to the existing LOSO and LOUO setups, average accuracy and edit score across the folds can be reported and used to compare models. However, examining each fold's performance and considering the relationship and similarity between the tasks in the training and test sets yields insights about the generalizability of the model to unseen tasks and the data needed to train a model.

D. Task Combination for Training

The unified set of finer-grained MP labels enable combining data from different tasks across datasets which can improve the diversity and size of training data and model generalization. On the other hand, the gesture labels are specific to each dataset and only tasks with similar labels within that dataset can be combined. To evaluate the effect of label granularity on task generalization, we use data from different combinations of tasks in the aggregated datasets for model training in both LOUO and LOTO setups. Using MPs, there were two combinations with similar context: S + NP = 'SNP' where both tasks have a task-specific needle state, and PT + PaS = 'PTPaS' where both tasks have a task-specific block state. Tasks could also be grouped together if they come from the same dataset: S + NP + KT = 'JIGSAWS' and PaS + PoaP = 'ROSMA'. Combining all of the data to train a model was referred to as 'All'. With gestures, only the SNP and JIGSAWS combinations could be used. For LOTO, we also considered specific combinations of data that tested on one task but removed the contextually similar tasks (defined above) from the training set to assess the importance of augmenting the training set with data from similar tasks.

E. Evaluation Metrics

We use the standard metrics accuracy, edit score [28], and mean average precision (mAP) [41] for the evaluation of gesture and MP recognition models. Micro mAP is reported for each verb to account for class imbalance.

V. Experimental Results

Experiments were performed on a computer with an Intel Core i9 CPU @ 3.60GHz and 64GB RAM, running Linux Ubuntu 18.04 LTS, and an NVIDIA GeForce RTX 2070 GPU running CUDA 10.2, and the models were built and trained using Torch 1.10.1 [42].

A. Gesture vs. Motion Primitive Recognition

In this section we present the performance of TCN models in recognizing gestures and MPs in comparison to state-of-the-art models and with different combinations of data.

Tables III and IV compare the accuracies and edit scores averaged over the folds of the LOUO setup for the TCN models trained to recognize gestures and MPs, respectively. Accuracies for two state-of-the-art models are also presented in Table III against which our TCN model performs comparably or better. The TCN performed best on S alone achieving an accuracy of 84.6% and an edit score of 87.7 which is also slightly better than the 79.6% accuracy and 85.8 edit score reported by [28] and comparable to the results of [22] for the TCN using only kinematic data (not shown in Table III).

TABLE III: Gesture recognition performance under the LOUO cross validation setup compared to state-of-the-art models using only kinematic data. Results for the state-of-the-art models were only available for the JIGSAWS tasks.

Tasks	Gestures			Baselines	
	Acc (%)	Edit Score	mAP	Acc (%)	Model
PT	73.5	83.8	80.7		
S	84.6	87.7	86.0	90.2	MS-RNN [34]
NP	78.4	85.2	86.4	75.3	SC-CRF [43]
KT	84.4	85.4	89.8	78.9	SC-CRF [43]
SNP	81.4	85.2	85.1		
JIGSAWS	80.9	82.0	85.7		

Despite KT only sharing two similar gestures and having a different task-specific context than the other two JIGSAWS tasks, the TCN's performance on KT is comparable to its performance on S (accuracy of 84.4%, edit score of 85.4). When data from multiple tasks is combined for the 'SNP' and 'JIGSAWS' models, the TCN models' accuracies are only about the average of their performances on individual tasks while the edit score for the JIGSAWS model drops to 82.0 which is lower than any single task in that dataset. Thus, there does not appear to be much benefit to combining data from the JIGSAWS tasks at the gesture level. The PoaP and PaS tasks from the ROSMA dataset did not have gesture labels, so no gesture recognition models were trained for them. The PT task of the DESK dataset did have gesture labels although their definitions were much closer in scope to MPs rather than the more complex gestures of the JIGSAWS dataset. The TCN only achieves an accuracy of 73.5% for gesture recognition on the PT task which is comparably lower than the performance of any of the MP recognition models for this task in the LOUO setup shown in Table IV. For the JIGSAWS tasks, the gesture recognition models performed much better than MP recognition models (only considering verbs). This suggests that the definitions and granularity of the labels in the surgical hierarchy affect activity recognition performance.

By examining Table IV, we note that MP recognition performance is better for the task in the DESK dataset, and to a somewhat lesser extent for tasks in the ROSMA dataset, than

TABLE IV: MP recognition performance with different task combinations under the LOUO cross validation setup.

Tasks	MPs		Left MPs		Right MPs	
	Acc	Edit	Acc	Edit	Acc	Edit
S	52.6	58.5	**66.0**	**65.2**	60.3	61.8
NP	52.3	53.1	**64.7**	**60.0**	55.9	54.8
KT	62.9	58.0	**71.2**	**67.2**	64.6	59.9
SNP	55.2	56.2	**66.5**	**62.2**	59.5	61.1
JIGSAWS	55.8	55.3	**66.4**	**63.5**	61.7	60.1
PoaP	67.4	74.6	**79.6**	72.6	79.3	**74.7**
PaS	70.2	76.5	**80.0**	**77.6**	78.5	75.9
ROSMA	67.5	**74.9**	**78.8**	73.1	78.2	73.6
PT	75.3	79.9	81.1	81.8	**82.0**	**82.4**
PTPaS	70.3	76.4	78.5	**77.8**	**78.8**	77.4
All	65.9	69.6	**75.0**	70.3	73.1	**70.7**

TABLE V: Number of examples (#) and mean average precision (mAP) of MPs for models trained on different combinations of tasks in the LOUO setup with micro mAP for all verbs (weighted by number of samples in each class).

Tasks	Grasp		Release		Touch		Untouch		Pull		Push		All verbs	
	#	mAP	#	mAP	#	mAP	#	mAP	#	mAP	#	mAP	#	mAP
S	471	57.6	441	48.7	518	58.1	314	27.6	194	72.2	179	55.1	2117	52.5
NP	373	63.0	365	57.0	330	57.0	206	16.2	114	69.1	119	34.2	1507	52.0
KT	283	64.5	247	69.1	135	43.8	111	18.6	235	85.3	0	N/A	1011	62.7
SNP	844	61.3	806	54.8	848	58.0	520	21.2	308	70.0	298	47.3	3624	52.9
JIGSAWS	1127	62.2	1053	58.7	983	53.0	631	20.7	543	72.6	298	41.5	4635	53.7
PoaP	577	52.8	556	55.3	1782	88.0	1261	47.2	525	58.3	2	33.5	4703	65.5
PaS	824	50.2	776	50.3	1598	88.9	1131	45.7	0	N/A	0	N/A	4329	63.3
ROSMA	1401	50.7	1332	53.1	3380	89.2	2392	45.3	525	59.2	2	5.1	9032	64.5
PT	323	48.3	313	61.1	539	90.3	364	68.3	0	N/A	0	N/A	1539	70.3
PTPaS	1147	48.7	1089	54.6	2137	89.8	1495	53.0	0	N/A	0	N/A	5868	65.9
All	2851	54.5	2698	55.4	4902	79.5	3387	43.5	1068	65.7	300	37.7	15206	60.7

for tasks in the JIGSAWS dataset. This could be because the JIGSAWS tasks (S, NP, KT) are more challenging with more complex grammar graphs [40], while the tasks in the ROSMA and DESK datasets are variations of a pick and place task with simpler grammar graphs. This is supported by the higher edit scores for the models trained on the ROSMA and DESK datasets than the models on the JIGSAWS dataset. Combining data at the MP level also resulted in performance metrics that are about the average of the individual tasks that were combined. But, training separate models for each side of the robot resulted in higher accuracies with comparable or better edit scores. So, having separate annotations and models for the left and right arms of the robot can improve MP recognition performance.

Furthermore, Table V shows the mAPs for each MP and micro average over all verbs for the MP recognition models in the LOUO set up. We note that class imbalance may have caused differences between the macro and micro mAPs for tasks from the DESK and ROSMA datasets where MPs with a greater number of instances sometimes had higher mAPs. None of these MP models perform as well as the gesture recognition models for the JIGSAWS tasks as listed in Table III, which achieve mAPs of up to 89.8. So additional work is needed to improve fine-grained activity recognition performance. Although the recognition models of [20] have been evaluated for verb recognition performance, a direct comparison to action triplet models is not fair as the data (kinematic vs. video) and tasks (robotic dry-lab vs. real laparoscopic surgery) are different.

B. Model Generalization

Table VI reports the accuracies and edit scores for models trained with different combinations of data in the LOTO setup and immediately shows limitations of existing gesture definitions. Note that only the JIGSAWS dataset had gesture labels that could be used in the LOTO setup, so gesture recognition models using tasks from different datasets could not be trained because gesture labels were not present or were not compatible. We observe that splitting the MP labels into separate transcripts and training separate models for the left and right arms of the robot generally results in improved accuracies compared to having a single model.

We find that a gesture recognition model trained on S or NP is able to transfer to NP or S, respectively, but when KT is added to the training set, performance is severely decreased. Specifically, a model tested on S drops from an accuracy of 48.5% to 24.4%, and a model tested on NP drops from 37.9% to 28.8% when KT is added to the training set. This is due to the lack of generalizable gesture labels between these tasks since S and NP have an almost completely different set of gestures than KT. Thus, gesture recognition for the KT task using a model trained on S and NP is particularly poor with an accuracy of only 6.8%. Hence, at the gesture level, combining data from different tasks is not beneficial for a model that must predict on an unseen task.

Comparatively, when MPs are used, the model is able to predict on a new task like KT by leveraging information learned from other tasks that are dissimilar to it such as S and NP. Adding data from a dissimilar task has a much smaller detrimental effect at the MP level than at the gesture level. For example, the model's accuracy drops less than 1% for S and 5% for NP when KT is added to the training set.

When the model must predict MPs on a dissimilar task with a different task-specific context state, then combining data from all tasks results in better performance compared to using only data in the same dataset. KT improves from an accuracy of 29.7% to 33.3% and PoaP improves from 54.8% to 56.5% by including data from other datasets.

For S and NP, we observe that models trained with data from the same dataset and with the same task-specific state variable perform better than models including data from the same dataset but without the same task-specific state variable. However, the opposite is true for PaS where models whose training sets included PoaP (same dataset) but not PT (same task-specific state variable) sometimes performed better.

For KT and PoaP, even though data with the same context was not available, models whose training sets included tasks from the same dataset generally performed better than models whose training sets did not. The poorest performing models for PaS were trained with data that only included PT, even though they had the same task-specific state variables. For PT, some

TABLE VI: MP and gesture recognition performance with different task combinations under LOTO cross validation setup.

Test Set	Training Set (Task combinations)						Gestures		MPs		Left MPs		Right MPs	
							Acc	Edit	Acc	Edit	Acc	Edit	Acc	Edit
S		NP	KT	PT	PaS	PoaP			39.0	**49.0**	62.3	**59.4**	42.9	**58.5**
S			KT	PT	PaS	PoaP			25.3	40.2	41.3	50.2	34.8	42.5
S		NP	KT				24.4	33.9	43.2	48.3	56.2	52.7	46.3	48.2
S		NP					**48.5**	**70.5**	**44.0**	47.7	**62.7**	58.7	**50.1**	54.7
NP	S		KT	PT	PaS	PoaP			40.8	48.5	**54.1**	**55.9**	41.6	46.4
NP			KT	PT	PaS	PoaP			35.6	44.5	46.2	51.9	34.4	39.9
NP	S		KT				28.8	38.2	37.2	46.9	49.9	52.8	44.7	48.3
NP	S						**37.9**	**52.7**	**42.2**	**48.6**	52.2	51.9	**46.0**	**52.8**
KT	S	NP		PT	PaS	PoaP			**33.3**	40.2	47.2	**51.9**	**35.2**	39.8
KT				PT	PaS	PoaP			22.6	37.5	37.7	36.5	25.1	36.8
KT	S	NP					**6.8**	**9.3**	29.7	**40.5**	**48.1**	50.4	34.5	**42.8**
PT	S	NP	KT		PaS	PoaP			**53.1**	**48.0**	**55.9**	**42.0**	43.9	38.6
PT	S	NP	KT			PoaP			44.5	44.4	49.0	37.6	**55.3**	**44.8**
PT					PaS				48.0	37.6	51.1	40.3	52.6	43.5
PaS	S	NP	KT	PT		PoaP			58.1	**65.5**	58.8	**60.5**	61.1	**58.0**
PaS	S	NP	KT			PoaP			60.7	65.0	58.5	58.5	61.4	57.7
PaS				PT		PoaP			58.0	64.1	**65.8**	58.3	**63.9**	57.2
PaS				PT					**61.0**	37.5	42.5	54.6	55.0	42.9
PaS						PoaP			58.4	62.9	59.5	57.2	59.8	56.1
PoaP	S	NP	KT	PT	PaS				**56.5**	**64.2**	**59.1**	**50.7**	**58.5**	**49.8**
PoaP	S	NP	KT	PT					53.4	47.8	50.4	45.9	36.0	43.9
PoaP					PaS				54.8	63.1	57.8	44.9	58.0	45.2

models that included PaS (same task-specific state variable) performed better than those that did not. Since tasks from the same dataset were performed by the same subjects, models whose training sets included tasks from the same dataset are tested on different tasks performed by known subjects. This is somewhat similar to the Leave-One-Supertrial Out (LOSO) cross validation method where models are tested on unseen trials performed by known subjects. Models evaluated using the LOSO method perform better than those using the LOUO method which suggests that including data from the same subjects may improve model performance. However, additional data and tests would be needed to determine if it is this or another feature of the dataset that is responsible for the performance improvement. Additional evaluations are also needed to verify that MPs enable task generalization for other types of models such as transformers [25].

VI. Discussion and Conclusion

In summary, we compare the performance of activity recognition in a case study of TCN models at different levels of the surgical hierarchy, evaluate their generalizability to unseen users and tasks, and draw insights from the combinations of tasks used to train these models.

We find that gesture-level recognition models perform better than motion primitive-level recognition models under the LOUO cross validation method which is consistent with the observations of [31]. Our models achieve comparable or better accuracies than state-of-the-art in recognizing gestures (from JIGSAWS).

Using motion primitives, we combine data from different datasets, tasks, and subjects and find that having separate models for the left and right sides improves performance. We also introduce the Leave-One-Task-Out (LOTO) cross validation setup, and perform the first evaluation of a surgical activity recognition model in terms of its ability to generalize to an unseen task. When tested on a task from a specific dataset, the model performed better if data from other tasks in that dataset were included in training. Also, models for tasks with different task-specific state variables perform best when data for all other tasks is aggregated for their training. Similarly, [44] evaluated the performance of surgeme classification models in sim2real domain transfer using different data percentages in the target domain and found that this improved the accuracies of their models. Thus, improved performance may be achieved by including a small percentage of data from the target test task in the training dataset.

Future work will focus on evaluating the task generalization of other state-of-the-art recognition models (e.g., recurrent neural networks and transformers) using both kinematic and vision data as well as other tasks and datasets.

Acknowledgment

This work was supported in part by the National Science Foundation grants DGE-1842490, DGE-1829004, and CNS-2146295 and by the Engineering-in-Medicine center at the University of Virginia. We thank Dr. Schenkman, Dr. Cantrell, and Dr. Chen for their medical feedback.

Competing Interests

The authors have no competing interests to declare that are relevant to the content of this article.

References

[1] D. Neumuth *et al.*, "Modeling surgical processes: A four-level translational approach," *Artificial intelligence in medicine*, vol. 51, no. 3, pp. 147–161, 2011.

[2] F. Lalys and P. Jannin, "Surgical process modelling: a review," *International journal of computer assisted radiology and surgery*, vol. 9, no. 3, pp. 495–511, 2014.

[3] L. Tao *et al.*, "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.

[4] B. Varadarajan *et al.*, "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.

[5] 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.

[6] M. S. Yasar and H. Alemzadeh, "Real-time context-aware detection of unsafe events in robot-assisted surgery," in *2020 50th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 2020, pp. 385–397.

[7] K. Hutchinson *et al.*, "Analysis of executional and procedural errors in dry-lab robotic surgery experiments," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 18, no. 3, p. e2375, 2022.

[8] Z. Li, K. Hutchinson, and H. Alemzadeh, "Runtime detection of executional errors in robot-assisted surgery," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE Press, 2022, p. 3850–3856.

[9] M. Ginesi, N. Sansonetto, and P. Fiorini, "Overcoming some drawbacks of dynamic movement primitives," *Robotics and Autonomous Systems*, vol. 144, p. 103844, 2021.

[10] N. Valderrama *et al.*, "Towards holistic surgical scene understanding," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2022, pp. 442–452.

[11] B. van Amsterdam, M. Clarkson, and D. Stoyanov, "Gesture recognition in robotic surgery: a review," *IEEE Transactions on Biomedical Engineering*, 2021.

[12] K. Hutchinson *et al.*, "Compass: a formal framework and aggregate dataset for generalized surgical procedure modeling," *International Journal of Computer Assisted Radiology and Surgery*, pp. 1–12, 2023.

[13] R. DiPietro *et al.*, "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.

[14] A. Goldbraikh *et al.*, "Using open surgery simulation kinematic data for tool and gesture recognition," *International Journal of Computer Assisted Radiology and Surgery*, pp. 1–15, 2022.

[15] G. Menegozzo *et al.*, "Surgical gesture recognition with time delay neural network based on kinematic data," in *2019 International Symposium on Medical Robotics (ISMR)*. IEEE, 2019, pp. 1–7.

[16] G. T. Gonzalez *et al.*, "From the dexterous surgical skill to the battlefield—a robotics exploratory study," *Military medicine*, vol. 186, no. Supplement_1, pp. 288–294, 2021.

[17] G. De Rossi *et al.*, "A first evaluation of a multi-modal learning system to control surgical assistant robots via action segmentation," *IEEE Transactions on Medical Robotics and Bionics*, 2021.

[18] D. Meli and P. Fiorini, "Unsupervised identification of surgical robotic actions from small non-homogeneous datasets," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 8205–8212, 2021.

[19] L. Li *et al.*, "Sirnet: Fine-grained surgical interaction recognition," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4212–4219, 2022.

[20] C. I. Nwoye *et al.*, "Rendezvous: Attention mechanisms for the recognition of surgical action triplets in endoscopic videos," *Medical Image Analysis*, vol. 78, p. 102433, 2022.

[21] Y. Qin *et al.*, "Temporal segmentation of surgical sub-tasks through deep learning with multiple data sources," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 371–377.

[22] B. Van Amsterdam *et al.*, "Gesture recognition in robotic surgery with multimodal attention," *IEEE Transactions on Medical Imaging*, 2022.

[23] N. Yong, P. Grange, and D. Eldred-Evans, "Impact of laparoscopic lens contamination in operating theaters: a study on the frequency and duration of lens contamination and commonly utilized techniques to maintain clear vision," *Surgical Laparoscopy Endoscopy & Percutaneous Techniques*, vol. 26, no. 4, pp. 286–289, 2016.

[24] J. C. Allers *et al.*, "Evaluation and impact of workflow interruptions during robot-assisted surgery," *Urology*, vol. 92, pp. 33–37, 2016.

[25] C. Shi, Y. Zheng, and A. M. Fey, "Recognition and prediction of surgical gestures and trajectories using transformer models in robot-assisted surgery," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 8017–8024.

[26] Y. Gao *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.

[27] I. Rivas-Blanco *et al.*, "A surgical dataset from the da vinci research kit for task automation and recognition," *arXiv preprint arXiv:2102.03643*, 2021.

[28] C. Lea *et al.*, "Temporal convolutional networks: A unified approach to action segmentation," in *European Conference on Computer Vision*. Springer, 2016, pp. 47–54.

[29] T. Neumuth *et al.*, "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] N. Madapana *et al.*, "Desk: A robotic activity dataset for dexterous surgical skills transfer to medical robots," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 6928–6934.

[31] A. Huaulmé *et al.*, "Micro-surgical anastomose workflow recognition challenge report," *Computer Methods and Programs in Biomedicine*, vol. 212, p. 106452, 2021.

[32] Y. Long *et al.*, "Relational graph learning on visual and kinematics embeddings for accurate gesture recognition in robotic surgery," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 13 346–13 353.

[33] M. Wagner *et al.*, "Comparative validation of machine learning algorithms for surgical workflow and skill analysis with the heichole benchmark," *Medical Image Analysis*, vol. 86, p. 102770, 2023.

[34] Y. A. Farha and J. Gall, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3575–3584.

[35] J. Zhang *et al.*, "Symmetric dilated convolution for surgical gesture recognition," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2020: 23rd International Conference, Lima, Peru, October 4–8, 2020, Proceedings, Part III 23*. Springer, 2020, pp. 409–418.

[36] A. Huaulmé *et al.*, "Peg transfer workflow recognition challenge report: Does multi-modal data improve recognition?" *arXiv preprint arXiv:2202.05821*, 2022.

[37] C. I. Nwoye *et al.*, "Cholectriplet2021: A benchmark challenge for surgical action triplet recognition," *Medical Image Analysis*, p. 102803, 2023.

[38] D. Itzkovich *et al.*, "Generalization of deep learning gesture classification in robotic-assisted surgical data: from dry lab to clinical-like data," *IEEE Journal of Biomedical and Health Informatics*, 2021.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[40] N. Ahmidi *et al.*, "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.

[41] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[42] A. Paszke *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in neural information processing systems*, vol. 32, 2019.

[43] C. Lea, G. D. Hager, and R. Vidal, "An improved model for segmentation and recognition of fine-grained activities with application to surgical training tasks," in *2015 IEEE winter conference on applications of computer vision*. IEEE, 2015, pp. 1123–1129.

[44] M. M. Rahman *et al.*, "Transferring dexterous surgical skill knowledge between robots for semi-autonomous teleoperation," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2019, pp. 1–6.