Event-Triggered Surgical Gesture Recognition with Change-Point Estimation

Alec Cotton[1], Georges Azzie[2], Ahmed Nasr[3], and Carlos Rossa[1]
[1]*Department of Systems and Computer Engineering, Carleton University, Ottawa, ON, Canada*
[2]*Department of Surgery, Hospital for Sick Children, Toronto, ON, Canada.*
[3]*Division of Paediatric Surgery, Children's Hospital of Eastern Ontario, Ottawa, ON, Canada.*

***Abstract*—Surgical gesture segmentation and classification algorithms are core components of modern surgical simulators. However, most existing algorithms operate off-line on pre-recorded data and require access to complete data sequences to segment them. Yet, they still underperform when kinematic data presents smooth transitions between gestures of unknown boundaries. This paper introduces a change point estimation (CPE) algorithm that detects shifts in statistical trends of cumulative feature vectors, identifies the start and end of a gesture, and then classifies the resulting segments. Unlike prior methods that rely on full-sequence processing or per-frame predictions, the proposed algorithm works on streamed data and achieves comparable accuracy to state-of-the-art methods that operate off-line on full kinematic sequences, within a 5 to 7% accuracy difference. The proposed CPE method is validated on three surgical tasks and 15 gestures from the JIGSAWS dataset using Leave-One-User-Out validation. CPE combined with a Random Forest classifier achieves a micro-accuracy comparable to manual classification when averaged across all tasks. We also show that integrating CPE with feature-based classification improves performance relative to benchmarks and state-of-the-art methods. This is the first application of CPE to surgical gesture recognition on streamed data, which can enable real-time gesture recognition in surgical simulators, supporting formative feedback and adaptive training.**

***Index Terms*—Gesture recognition, surgical robotics, surgical simulation, medical education.**

I. Introduction

Surgical simulators provide a controlled environment for practicing and honing procedural skills, while recording kinematic and force data from surgical instruments for assessment of surgical performance [1]–[6]. A fundamental building block of modern surgical simulators is surgical gesture recognition; the process of identifying fundamental movements or actions from measured data [7]–[12]. In surgical tasks, defined as a sequence of gestures executed in a predefined order, gesture recognition comprises segmenting time series into meaningful subintervals [13]–[15], and then classifying each identified segment as a predefined gesture class [16]–[18].

Email: aleccotton@carleton.ca (A. Cotton); georges.azzie@sickkids.ca (G. Azzie); anasr@cheo.on.ca (A. Nasr), carlos.rossa@carleton.ca (C. Rossa).

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC), [funding reference number RGPIN-2025-05468]. Cette recherche a été financée par le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG), [numéro de référence RGPIN-2025-05468].

Segmentation methods must discern statistically different intervals in time-series and identify transitions between them [19]–[24]; a task that is easier when the entire surgical task is available as the algorithm can exploit temporal dependencies over time. For example, hidden Markov models and dynamic time warping assign class labels to segments by mapping time steps to the most likely class only after the entire time sequence has been observed [25]–[27]. Similarly, many deep learning models require access to the complete time sequence to produce per-time-step predictions [13].

While offline segmentation and classification are effective to provide summative feedback once training is completed, online classification would enable simulators to deliver formative feedback during task execution, potentially improving training outcomes [7], [9]. However, online classification is substantially more challenging, as the segmentation algorithm must process incoming data incrementally without access to future statistical trends. Segmentation algorithms can generally be categorized as unsupervised or supervised. In the former, segment bounds are inferred by detecting statistically distinct regions in the data without relying on labeled data. Supervised segmentation requires labeled data. It determines segment boundaries based on the algorithm's confidence in assigning segments to a predefined gesture class. Supervised methods are often preferred as the labels generally correspond to interpretable actions.

Several methods have been proposed to segment temporal sequences on streamed data. For example, supervised Gaussian mixture models learn class-specific clusters and assign incoming samples to one of them [28]. Deep learning methods, such as temporal convolution networks (tCN) [29] and transformer networks [30], make predictions at each time step with only current and past time steps. Convolutional neural networks or recurrent neural networks extract spatial features from data before identifying temporal dependencies between them [31], [32]. Cumulative Sum Control Chart monitors the cumulative sum of a quality index and signals a segment boundary when the running total surpasses a learned threshold [33]. In surgical gesture recognition, human-annotated boundaries often occur between gestures that transition smoothly and exhibit minimal statistical change. As a result, such methods can lead to over-segmentation, producing segments that have no statistical

meaning, or under-segmentation, missing or delaying actual gesture boundaries.

A potential solution to improve event-triggered gesture segmentation from smooth, continuous time sequences is to extract quantitative features from the data using a sliding time window, track their evolution over time statistically, and identify changes in their trend that indicate gesture boundaries. The same accumulated features can then be used for both segmentation and classification, thereby combining them under a single feature set. Change point estimation (CPE) is a method that locates points in a time series where the statistical distribution of the data changes, and thus can be used to divide the sequence into segments that differ statistically, independent of their duration. CPE has been used in speech recognition to identify boundaries between speakers [34], in computational cardiology to identify different phases in the heart beat cycle [23], and in telecommunications for network monitoring [35].

In this paper, we introduce CPE to segment surgical gestures in an event-driven manner using a sliding window of cumulative features extracted from streamed kinematic data. The proposed algorithm identifies changes in these features over time to determine the beginning and end of an unknown gesture, after which the segment between two detected transition points is classified via interrupts. Unlike the aforementioned methods, data are not classified until a change point is detected, avoiding classifying incomplete segments when operating on streamed data. The ability to identify the boundaries of a surgical gesture during task execution can enable surgical simulators to provide formative feedback, substantially improving the outcomes of simulation training. We implement and evaluate the proposed algorithm using the Leave-One-User-Out (LOUO) benchmark using an open source database acquired on the da Vinci surgical system [36], [37]. When operating on streamed data, the proposed algorithm achieves accuracy comparable to offline state-of-the-art methods that use full kinematic sequences, within a 5–7% margin. To the best of the authors' knowledge, this is the first implementation of CPE to surgical gesture recognition on streamed data.

The remainder of the paper is structured as follows: Section II provides an overview of the statistical feature encoding used for both CPE and classification. Section III details the proposed CPE algorithm. Section IV describes the dataset and experimental setup, followed by results and discussion in Section V. Section VI provides conclusions.

II. Statistical Feature Encoding

Consider a dataset of g time series $\mathbf{X} \in \mathfrak{R}^{n\times h}$, each of which comprises h column vectors ${}^{i}\mathbf{x} \in \mathfrak{R}^{n\times 1}$, with $1 \le i \le h$, each representing a temporal sequence of length n:

$$\mathbf{X} = [{}^{1}\mathbf{x}, {}^{2}\mathbf{x}, \ldots, {}^{h}\mathbf{x}]. \tag{1}$$

Let $X_{i,j}$ be an element in the i^{th} column and j^{th} row of $\mathbf{X}$. We extract the mean (μ), variance (σ^2), skew (γ), and kurtosis (κ) from a window of length k on each column. For a window $a \le j \le b$ with $k = b - a$. The mean, given by

$$\mu_i = \frac{1}{k}\sum_{j=1}^{k}\left(X_{j,i}\right), \tag{2}$$

represents the average value of each feature in the gesture over time. The variance provides information regarding the degree to which each feature fluctuates over a gesture, given by:

$$\sigma_i^2 = \frac{1}{k-1}\sum_{j=1}^{k}(X_{j,i} - \mu_i)^2. \tag{3}$$

The skewness quantifies the asymmetry of this variation from the mean and is expressed as:

$$\gamma_i = \frac{k}{(k-1)(k-2)}\sum_{j=1}^{k}\left(\frac{X_{j,i} - \mu_i}{\sigma_i}\right)^3, \tag{4}$$

and the kurtosis denotes each feature's propensity for experiencing extreme events, computed as:

$$\kappa_i = \frac{k(k+1)}{(k-1)(k-2)(k-3)}\sum_{j=1}^{k}\left(\frac{X_{j,i} - \mu_i}{\sigma_i}\right)^4 - \frac{3(k-1)^2}{(k-2)(k-3)}. \tag{5}$$

We compute these four features on each of the h columns of $\mathbf{X}$ to form a statistical feature matrix ${}^{j}\mathbf{H} \in \mathfrak{R}^{h\times 4}$:

$${}^{j}\mathbf{H} = \begin{bmatrix} \mu_1 & \mu_2 & \ldots & \mu_h \\ \sigma_1^2 & \sigma_2^2 & \ldots & \sigma_h^2 \\ \gamma_1 & \gamma_2 & \ldots & \gamma_h \\ \kappa_1 & \kappa_2 & \ldots & \kappa_h \end{bmatrix}. \tag{6}$$

This constant-sized matrix represents the 4 Pearson statistical moments within the window and serves as the input to both the CPE algorithm and the gesture classifier. The feature matrix ${}^{\mathbf{j}}\mathbf{H}$ is flattened into a feature vector ${}^{\mathbf{j}}\mathbf{h}$ in order to be processed in subsequent steps. For the sliding-window regressor used in CPE, to make the "before vs. after" contrast explicit, the feature vector is computed separately on each half of the window and concatenated, along with the per-column mean absolute difference between the two halves. Specifically, for a window of length k with first half $W^{(1)}$ and second half $W^{(2)}$, the window feature vector ${}^{j}\tilde{\mathbf{h}} \in \mathbb{R}^{9h}$ is denoted by ${}^{\mathbf{j}}\tilde{\mathbf{h}}$:

$${}^{j}\tilde{\mathbf{h}} = \left[\mathrm{vec}\left({}^{j}\mathbf{H}^{(1)}\right);\ \mathrm{vec}\left({}^{j}\mathbf{H}^{(2)}\right);\ \left|\bar{\mu}^{(2)} - \bar{\mu}^{(1)}\right|\right], \tag{7}$$

where ${}^{j}\mathbf{H}^{(l)}$ denotes the feature matrix computed on half l, $\bar{\mu}^{(l)}$ is the column-wise mean of $W^{(l)}$, and vec$(\cdot)$ denotes column-wise flattening. Feature vectors used in segment classification represent the accumulation of new time steps in a constant-sized feature vector given by: ${}^{\mathbf{j}}\bar{\mathbf{h}}$:

$${}^{\mathbf{j}}\bar{\mathbf{h}} = \mathrm{vec}({}^{\mathbf{j}}\mathbf{H}) \tag{8}$$

As the flattened feature vector is updated in over incoming data streamed continuously, the CPE algorithm analyzes the trend of these features over time to detect transitions between gestures before the segment between two change points is classified. This process is described in the next section.

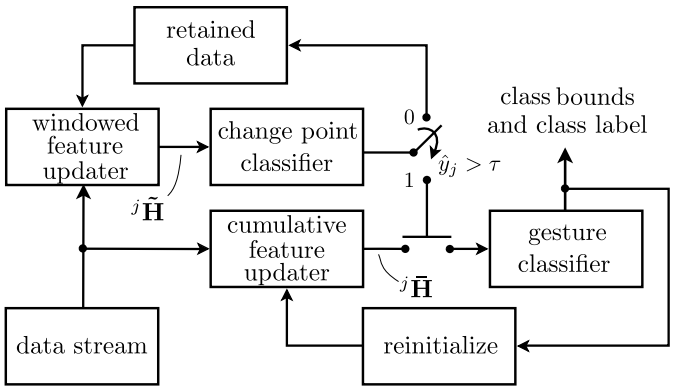


Fig. 1. Event-triggered gesture segmentation and classification. The cumulative feature matrix is only evaluated on the gesture classifier when a change point is detected above threshold h.

III. Change-Point Detection for Gesture Segmentation

To identify change points, we simultaneously compute and update two feature vectors:

1) A fixed-length sliding window feature vector $^j\tilde{\mathbf{h}}$ given by (7) is used to identify change points at each time step;
2) An increasing-length cumulative feature matrix $^j\bar{\mathbf{h}}$ is updated from the last detected change point and is used for gesture classification.

As illustrated in Fig. (1), both statistical feature vectors are updated concurrently. When a change point is detected, $^j\bar{\mathbf{h}}$ is classified and reinitialized, and the process restarts.

The proposed method employs two models. The first is a change point regressor:

$$R(^j\tilde{\mathbf{h}}) \in (0, 1), \tag{9}$$

which outputs a scalar indicating the likelihood that a change point is centred within the window $^j\tilde{\mathbf{h}}$. The regression target during training is a triangular kernel that assigns higher scores to windows whose centre coincides with a true gesture boundary:

$$R(^j\tilde{\mathbf{h}}) \leftarrow 1 - \frac{|t - 0.5k|}{0.5k}, \tag{10}$$

where t denotes the time step of a given true change point and k is the window width. A Random Forest regressor f_R is trained on labeled instances of $^j\tilde{\mathbf{h}}$.

The regressor output is interpreted using a Cumulative Sum (CUSUM) accumulator [33]. At each time step t, the accumulator S_t is updated as:

$$S_t = \max\left(0,\ S_{t-1} + R(^t\tilde{\mathbf{h}}) - k_{\text{cusum}}\right), \tag{11}$$

where k_{cusum} is a drift allowance parameter that suppresses low-level noise. A change point is declared at time t when $S_t \geq h$ and at least $d_{\min}$ frames have elapsed since the last change point, after which S_t is reset to zero:

$$\text{CP at } t : S_t \geq h \ \text{ and } \ t - t_{\text{last}} \geq d_{\min}. \tag{12}$$

This guarantees that the decision at step t relies only on regressor outputs up to and including t. The parameters k_{cusum} and h are optimised jointly by grid search on a held-out validation trial from the training set at each fold, maximising segment-level F1@25. This use of a validation split ensures that the CUSUM policy is learned from training data without accessing the test fold.

The second model is a gesture classifier:

$$C_s(^j\bar{\mathbf{h}}) \rightarrow \text{ARGMAX}\{c_1, \ldots, c_N\}, \tag{13}$$

trained gesture segments obtained from validation splits during segmentation training. The training policy is outlined in the following subsection. To make the classifier robust to imperfect segment boundaries produced by CPE at test time, the training set is augmented with boundary-jittered copies of each gesture: for each true segment $[s, e]$, additional samples are generated by shifting s and e independently by a random amount up to $\pm\lfloor\delta(e-s)\rfloor$ frames, where $\delta = 0.15$. On streamed test data, this model maps the feature matrix $^j\bar{\mathbf{H}}$ accumulated between two detected change points to a gesture label.

IV. Dataset and Testing

To validate the proposed algorithm, we use the John Hopkins University and Intuitive Surgical Gesture and Skill Assessment Working Set (JIGSAWS) dataset collected on the da Vinci surgical robotic system. The dataset contains instances of three surgical tasks, i.e., knot-tying, needle-passing, and suturing. Each task consists of a sequence of up to 15 surgical gestures executed in a particular sequence, as summarized in Table I. Eight surgeons of varying skill levels performed each task five times. Kinematic data of the two laparoscopic tools was recorded at 30 Hz, i.e., 3 Cartesian positions, 3 orientations, and 3 linear and 3 angular velocities, in addition to gripper angle (13 variables for each tool), giving $h = 26$ columns per instance.

We validate the proposed method on all three tasks under the LOUO validation protocol, as it is the most commonly reported benchmark and represents the more challenging generalisation setting: training is performed on all but one user, who is left out for testing on each fold, yielding eight folds in total. The goal is to classify gestures on streamed data with unknown bounds. We omit gesture 10 (G10) due to the associated heavy class imbalance; there are only five instances of G10, which is significantly lower than the number of instances of all other gestures as seen in Table I. To evaluate the algorithm, we perform the following experiments:

1) **Baseline (BL)**: Pre-segmented classification with known gesture bounds, establishing the performance ceiling for feature-based classification.
2) **Proposed method (F+CPE)**: Using streamed data, the proposed CPE method is applied using a window size of $k = 30$ to detect gesture boundaries, after which a Random Forest classifier is applied to each detected segment.

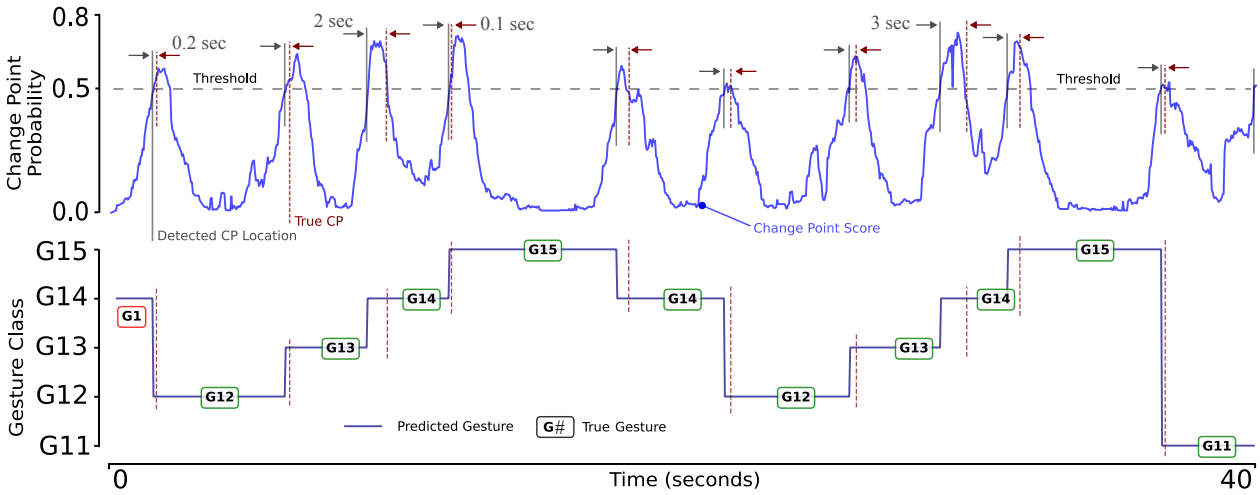


Fig. 2. Sample segmentation and classification result of a suturing task. The top panel shows the predicted change point (CP) score given by (10). A CP is declared when the CUSUM accumulator exceeds threshold h. The true gesture boundaries (true CP) are also indicated. The bottom panel shows the predicted *vs* true gesture labels between detected CPs using the Random Forest classifier.

TABLE I
GESTURE IN THE DATASET, GESTURES IN EACH TASK, AND NUMBER OF SAMPLES (# OF S)

Gesture	**Action**	**# of S**
G1	Reaching for needle with right hand	78
G2	Positioning needle	283
G3	Pushing needle through tissue	275
G4	Transferring needle from left to right	202
G5	Moving to center with needle in grip	68
G6	Pulling suture with left hand	275
G8	Orienting needle	76
G9	Using right hand to help tighten suture	25
G10	Loosening more suture	5
G11	Dropping suture and moving to end point	100
G12	Reaching for needle with left hand	70
G13	Making C loop around right hand	75
G14	Reaching for suture with right hand	98
G15	Pulling suture with both hands	73
Task	**Gestures in Task**	
Suturing	G1–6, G8–11	40
Knot-tying	G1, G11–15	38
Needle passing	G1–6, G8, G11	40

V. RESULTS AND DISCUSSION

Fig. 2 shows a sample segmentation and classification result of a suturing task. The top panel shows the predicted change point score given by (10), with change points declared according to (12). The bottom panel shows the gesture label assigned to segments between change points alongside the true gesture class.

A. Classification Results

Fig. 3 shows the confusion matrices for each task, summarizing classification results for the **BL** (left column) and **F+CPE** tests (right column). Results are reported per frame, with total of 273,553 temporal frames of kinematic data classified in order to construct these confusion matrices. We observe that the quality of classification for each gesture in a task changes from the **BL** experiment to the **F+CPE** experiment. This is likely due to the discrepancy in time steps between gestures, as well as the frequency of each gesture class. As outlined in [22], [38], the proportion of manually annotated time steps on the JIGSAWS dataset is between 75 – 80%. Thus, a gesture classification score with unknown bounds above 80% is deemed suitable for simulation-based training.

Table II(a) summarises the performance of the Random Forest classifier on the LOUO suturing task for both the pre-segmented baseline and the proposed F+CPE method. Feature-based classification with known bounds establishes a strong performance ceiling; the classifier achieves a micro accuracy of 89.30% and a macro precision of 90.03% on pre-segmented gesture classification, averaged across all surgical tasks. The addition of CPE to segment streamed data prior to classification offers performance comparable to that achieved with pre-segmented data. Random forest combined with CPE achieves a micro accuracy of 76.06% and a macro precision of 72.83% on joint segmentation and classification, demonstrating that online gesture recognition is achievable with only a modest reduction relative to the pre-segmented ceiling.

Notably, the segmentation scores for the needle passing task are worse than suturing and knot tying scores, as seen in Table II. As a result, the classification scores for the needle passing task are lower in **F+CPE** relative to the corresponding **BL** result. This may imply that the difficulty in recognizing gestures depends on a model's ability to identify gesture boundaries. A significant decrease in model generalisation for the needle passing task is well documented in the literature [37]. In contrast, the suturing and knot tying task segmentation scores improve considerably (edit distances of 0.35 and 0.32 respectively), and only a 10 – 16% decrease in micro accuracy is observed between **BL** and **F+CPE**.

Table II(b) reports sequence-level metrics for the proposed F+CPE method. The average normalised edit distance of

0.353 and segment-level F1@10 of 74.2% indicate that the CPE algorithm successfully recovers the majority of gesture boundaries and their ordering, even when operating causally on streamed data.

B. Comparison with State-of-the-Art Models

The best achieved results on streamed, unsegmented surgical data (RF with CPE) are compared against results obtained from the literature in Table II(b). To the best of our knowledge, [39], [40], and [29] are the only papers reporting on streamed data. The vast majority of models in the literature operate off-line on entire sequences to segment and classify gestures; this is the case for SC-CRF [37], although the tCN [41] is reported to operate causally. As expected, online methods underperform relative to offline methods because they are constrained to per-frame information rather than processing the entire sequence beforehand. These results indicate the proposed method handles class imbalance well, as reflected in the macro accuracy and precision metrics relative to the online alternatives.

VI. Conclusion

Rather than classifying data per-frame, the proposed method detects change points using a feature vector computed over a sliding window prior to classifying intervals between them. Pearson moments prove to be an effective set of features for pre-segmented gesture classification, with a micro accuracy of 90.03% averaged across needle passing, knot tying, and suturing tasks. The addition of CPE makes it possible to use the Random Forest classifier when gesture boundaries are unknown, achieving a micro accuracy of 76.06% and a precision of 72.83% averaged across all tasks on streamed data, with an average normalised edit distance of 0.353 and F1@10 of 74.2%.

Combining CPE and classification improves classification results with respect to the state-of-the-art for unsegmented gesture sequences. Yet, several key issues must be addressed in future work. Training a classifier on gestures with known bounds, assuming that CPE will detect them, is a major assumption in this study. A modern machine learning pipeline could be devised to ensure that the classifier is trained on segments most likely to be detected by the segmentation model, while maximising CPE accuracy. The formulation of such a framework for training both classification and segmentation in tandem may lead to segmentation that matches more closely that of human annotations.

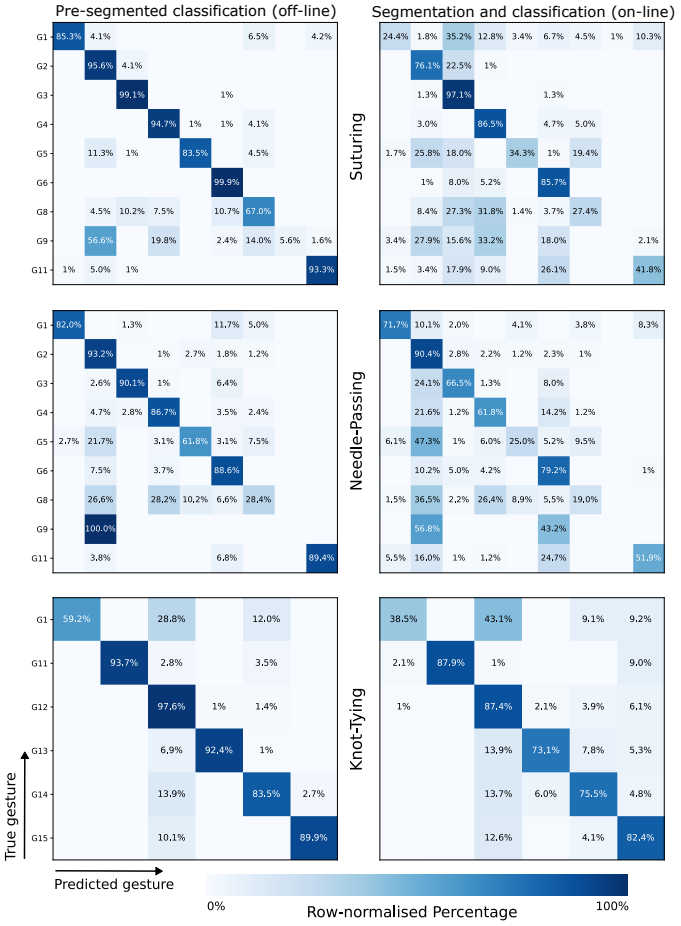


Fig. 3. Row-normalized confusion matrix of all time steps classified across Needle Passing, Knot Tying, and Suturing tasks under LOUO validation. Cell values show raw counts, while colour indicates the percentage normalized per predicted class (rows sum to 100%). Gesture frequencies vary across classes.

References

[1] A. Abasian *et al.*, "Stochastic Sequential Sensory Selection for Gesture Recognition in KineticoMyoGraphy Guided Bionic Hands," *IEEE Transactions on Medical Robotics and Bionics*, vol. 7, no. 1, pp. 325–336, 2025.

[2] Y. Hao, H. Zhang, Z. Zhang, C. Hu, and C. Shi, "Development of Force Sensing Techniques for Robot-Assisted Laparoscopic Surgery: A Review," *IEEE Transactions on Medical Robotics and Bionics*, vol. 6, pp. 868–887, Aug. 2024.

[3] G. G. Muscolo and P. Fiorini, "Force–Torque Sensors for Minimally Invasive Surgery Robotic Tools: An Overview," *IEEE Transactions on Medical Robotics and Bionics*, vol. 5, pp. 458–471, Aug. 2023.

[4] Z. Tang, S. Wang, M. Li, and C. Shi, "Development of a Distal Tri-Axial Force Sensor for Minimally Invasive Surgical Palpation," *IEEE Transactions on Medical Robotics and Bionics*, vol. 4, pp. 145–155, Feb. 2022.

[5] K. MacWilliams, J. Green, A. Nasr, G. Azzie, and C. Rossa, "Optimizing sensor selection in laparoscopic simulators: Lessons learned in a robotic platform," in *Int. Instrumentation and Measurement Technology Conf.*, pp. 1–6, 2025.

[6] Z. Wang and A. M. Fey, "Deep learning with convolutional neural network for objective skill evaluation in robot-assisted surgery," *International Journal of Computer Assisted Radiology and Surgery*, vol. 13, no. 12, pp. 1959–1970, 2018.

[7] H. Maya *et al.*, "To the point: medical education review of the role of simulators in surgical training," *American J. of Obstetrics and Gynecology*, vol. 199, no. 4, pp. 338–343, 2008.

[8] Y. Okuda *et al.*, "The utility of simulation in medical education: What is the evidence?," *Mount Sinai J. of Medicine*, vol. 76, no. 4, pp. 330–343, 2009.

[9] W. H. Ryu *et al.*, "Systematic review of patient-specific surgical simulation: Toward advancing medical education," *J. of Surgical Education*, vol. 74, no. 6, pp. 1028–1038, 2017.

[10] V. N. Palter and T. P. Grantcharov, "Simulation in surgical education," *CMAJ*, vol. 182, no. 11, pp. 1191–1196, 2010.

[11] M. Graafland, J. Schraagen, and P. Schijven, "Systematic review of serious games for medical education and surgical skills training," *British J. of Surgery*, vol. 99, pp. 1322–1330, 09 2012.

[12] W. McGaghie *et al.*, "A critical review of simulation-based medical education research," *Medical Education*, vol. 44, no. 1, pp. 50–63, 2010.

TABLE II

CLASSIFICATION RESULTS OF PRE-SEGMNETED (A) AND UNSEGMENTED GESTURES (B) IN SUTURING (ST), NEEDLE PASSING (NT) AND KNOT TYING (NT). MICRO: OVERALL ACCURACY (ACC); MACRO: MEAN PER-CLASS RECALL; PREC: MEAN PER-CLASS PRECISION (%); EDIT: NORMALISED EDIT DISTANCE; AND F1@k: SEGMENTAL F1 SCORE AT OVERLAP THRESHOLD k%. THE BEST MODEL FOR EACH PERFORMANCE METRIC IS INDICATED IN BOLD.

(a) Pre-segmented Gesture Classification (Baseline, off-line)

Method	Metric	ST	NP	KT
BL	Micro	**92.9**	**85.1**	**89.9**
	Macro	**80.4 ± 29.9**	68.9 ± 33.1	**86.1 ± 14.0**
	Prec	**93.6 ± 5.2**	**82.4 ± 15.8**	**94.1 ± 10.4**
GMM HMM [37]	Micro	80.8	66.2	78.4
	Macro	65.0 ± 33.0	62.7 ± 16.3	72.6 ± 21.3
	Prec (%)	71.9 ± 33.0	66.55 ± 25.4	83.2 ± 13.0
Soup CNN [42]	Micro	85.0	78.4	83.8
	Macro	65.6 ± 36.8	**71.5 ± 19.1**	82.7 ± 7.6
	Prec (%)	74.5 ± 29.2	73.7 ± 13.7	81.7 ± 10.5
BoF [37]	Micro	82.97	67.11	86.52
	Macro	67.4 ± 36.3	63.1 ± 19.9	85.4 ± 7.8

(b) Unsegmented Gesture Segmentation/Classification (online)

F+CPE (this paper)	Micro	76.3	72.3	**79.6**
	Macro	52.6 ± 34.3	51.7 ± 30.5	74.1 ± 18.5
	Prec	66.8 ± 26.3	68.5 ± 12.2	**83.2 ± 10.0**
	Edit	0.35 ± 0.05	0.39 ± 0.11	0.32 ± 0.13
	F1@10	75.5 ± 4.9	69.5 ± 11.2	77.6 ± 10.5
	F1@25	72.8 ± 4.4	65.9 ± 13.3	75.7 ± 11.5
	F1@50	66.3 ± 5.3	56.8 ± 12.3	68.4 ± 16.1
TDNN [39]	Micro	74.4 ± 7.4	64.3 ± 11.6	73.4 ± 9.5
	Macro	53.8 ± 8.4	48.2 ± 7.83	70.2 ± 12.1
	Prec	67.3 ± 9.5	55.5 ± 16.3	62.1 ± 21.6
	F1	0.75 ± 0.1	0.63 ± 0.15	0.72 ± 0.17
SC-CRF [37]	Micro	**81.7**	**74.7**	78.9
	Macro	**68.6 ± 29.8**	**66.4 ± 19.8**	**77.5 ± 10.8**
	Prec	**70.2 ± 26.9**	**71.3 ± 13.7**	76.2 ± 14.0
ST-CNN +SEG [29]	Acc	74.2	-	-
	Edit	33.4	-	-
C3DTra [40]	Acc	75.8	-	-
ST-TCN [41]	Acc	79.6	-	-
	Edit	**14.2**	-	-

[13] P. Hu *et al.*, "Real-time semantic segmentation with fast attention," *IEEE Robotics and Automation Letters*, vol. 6, no. 1, pp. 263–270, 2021.

[14] G. Ding, F. Sener, and A. Yao, "Temporal action segmentation: An analysis of modern techniques," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 2, pp. 1011–1030, 2024.

[15] J. Fard, S. Ameri, R. B. Chinnam, and E. Darin, "Soft boundary approach for unsupervised gesture segmentation in robotic-assisted surgery," *IEEE Robotics and Automation Letters*, vol. 2, no. 1, pp. 171–178, 2017.

[16] H. I. Fawaz *et al.*, "Deep learning for time series classification: a review," *Data Mining and Knowledge Discovery*, vol. 33, no. 4, pp. 917–963, 2019.

[17] A. Abanda, U. Mori, and J. Lozano, "A review on distance based time series classification," *Data Mining and Knowledge Discovery*, vol. 33, no. 2, pp. 378–412, 2019.

[18] J. Faouzi, "Time series classification: A review of algorithms and implementations," in *Machine Learning (Emerging Trends and Applications)* (K. Kotecha, ed.), p. 298, Proud Pen, 2024.

[19] S. Zolhavarieh, S. Aghabozorgi, and Y. W. Teh, "A review of subsequence time series clustering," *The Scientific World Journal*, no. 1, p. 312521, 2014.

[20] F. Despinoy *et al.*, "Unsupervised trajectory segmentation for surgical gesture recognition in robotic training," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1280–1291, 2016.

[21] S. Krishnan *et al.*, "Transition state clustering: Unsupervised surgical trajectory segmentation for robot learning," *The Inter. Journal of Robotics Research*, vol. 36, no. 13-14, pp. 1595–1618, 2017.

[22] N. Ahmidi *et al.*, "String motif-based description of tool motion for detecting skill and gestures in robotic surgery," in *Medical Image Computing and Computer-Assisted Intervention*, pp. 26–33, Springer, 2013.

[23] I. Beraza and I. Romero, "Comparative study of algorithms for ecg segmentation," *Biomedical Signal Processing and Control*, vol. 34, pp. 166–173, 2017.

[24] D. Wang and J. Chen, "Supervised speech separation based on deep learning: An overview," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.

[25] H. Cheng, Z. Dai, and Z. Liu, "Image-to-class dynamic time warping for 3d hand gesture recognition," in *Int. Conf. on Multimedia and Expo*, pp. 1–6, 2013.

[26] S. B. Wang, A. Quattoni, L.-P. Morency, D. Demirdjian, and T. Darrell, "Hidden conditional random fields for gesture recognition," in *Computer Society Conf. on Computer Vision and Pattern Recognition*, vol. 2, pp. 1521–1527, 2006.

[27] M. Elmezain *et al.*, "A hidden markov model-based continuous gesture recognition system for hand motion trajectory," in *Int. Conf. on Pattern Recognition*, pp. 1–4, 2008.

[28] B. v. Amsterdam, H. Nakawala, E. D. Momi, and D. Stoyanov, "Weakly supervised recognition of surgical gestures," in *Int. Conf. on Robotics and Automation*, pp. 9565–9571, 2019.

[29] C. Lea *et al.*, "Segmental Spatiotemporal CNNs for Fine-Grained Action Segmentation," in *Computer Vision*, vol. 9907, pp. 36–52, 2016. Series Title: Lecture Notes in Computer Science.

[30] A. D'Eusanio *et al.*, "A transformer-based network for dynamic hand gesture recognition," in *Int. Conf. on 3D Vision*, pp. 623–632, 2020.

[31] X. Gao *et al.*, "Transsvnet: Accurate phase recognition from surgical videos via hybrid embedding aggregation transformer," in *Medical Image Computing and Computer Assisted Intervention Conf.*, pp. 593–603, Springer, 2021.

[32] A. Twinanda *et al.*, "Endonet: a deep architecture for recognition tasks on laparoscopic videos," *IEEE Transactions on Medical Imaging*, vol. 36, no. 1, pp. 86–97, 2016.

[33] D. Wang, Y. Yu, and A. Rinaldo, "Univariate mean change point detection: Penalization, CUSUM and optimality," *Electronic Journal of Statistics*, vol. 14, no. 1, pp. 1917 – 1961, 2020.

[34] V. Gupta, "Speaker change point detection using deep neural nets," in *Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 4420–4424, 2015.

[35] E.-S. Apostol *et al.*, "Change point enhanced anomaly detection for iot time series data," *Water*, vol. 13, no. 12, 2021.

[36] Y. Gao *et al.*, "Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling," in *MICCAI workshop*, vol. 3, p. 3, 2014.

[37] 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.

[38] S. Vedula *et al.*, "Analysis of the structure of surgical activity for a suturing and knot-tying task," *PloS One*, vol. 11, no. 3, p. e0149174, 2016.

[39] G. Menegozzo, D. DallrAlba, C. Zandona, and P. Fiorini, "Surgical gesture recognition with time delay neural network based on kinematic data," in *Int. Symp. on Medical Robotics*, (Atlanta, GA, USA), pp. 1–7, IEEE, Apr. 2019.

[40] A. Gazis, P. Karaiskos, and C. Loukas, "Surgical Gesture Recognition in Laparoscopic Tasks Based on the Transformer Network and Self-Supervised Learning," *Bioengineering*, vol. 9, p. 737, Nov. 2022.

[41] C. Lea *et al.*, "Temporal Convolutional Networks: A Unified Approach to Action Segmentation," in *Computer Vision – ECCV Workshops* (G. Hua and H. Jégou, eds.), vol. 9915, pp. 47–54, Cham: Springer International Publishing, 2016.

[42] K. MacWilliams, A. Nasr, G. Azzie, J. R. Green, and C. Rossa, "Jigsaws soup: Improved laparoscopic surgical gesture classification, and skill level evaluation using model souping," *Computers in Biology and Medicine*, vol. 206, p. 111617, 2026.