Think Step by Step: Chain-of-Gesture Prompting for Error Detection in Robotic Surgical Videos

Zhimin Shao[1], Jialang Xu[2], Danail Stoyanov[2], *Fellow, IEEE*,
Evangelos B. Mazomenos[2], *Member, IEEE* and Yueming Jin[3], *Member, IEEE*

Abstract—**Despite advancements in robotic systems and surgical data science, ensuring safe execution in robot-assisted minimally invasive surgery (RMIS) remains challenging. Current methods for surgical error detection typically involve two parts: identifying gestures and then detecting errors within each gesture clip. These methods often overlook the rich contextual and semantic information inherent in surgical videos, with limited performance due to reliance on accurate gesture identification. Inspired by the chain-of-thought prompting in natural language processing, this letter presents a novel and real-time end-to-end error detection framework, Chain-of-Gesture (COG) prompting, integrating contextual information from surgical videos step by step. This encompasses two reasoning modules that simulate expert surgeons' decision-making: a Gestural-Visual Reasoning module using transformer and attention architectures for gesture prompting and a Multi-Scale Temporal Reasoning module employing a multi-stage temporal convolutional network with slow and fast paths for temporal information extraction. We validate our method on the JIGSAWS dataset and show improvements over the state-of-the-art, achieving 4.6% higher F1 score, 4.6% higher Accuracy, and 5.9% higher Jaccard index, with an average frame processing time of 6.69 milliseconds. This demonstrates our approach's potential to enhance RMIS safety and surgical education efficacy. The code will be available.**

Index Terms—**Surgical error detection, Video-language learning, Prompt engineering, Computer vision for medical robotics.**

I. Introduction

THE advent of robot-assisted minimally invasive surgery (RMIS) has revolutionized operative procedures across various medical specialties, from urology to general surgery. RMIS extends human dexterity, offering unprecedented precise instrument navigation and enabling vivid observation of surgical scene [1]–[3]. Despite clear advancements, RMIS requires a high level of proficiency for surgeons to master the manipulation of sophisticated robotic systems. The safety of RMIS can be inevitably compromised due to technical errors [4], [5], such as unintended instrument operation, alteration of the surgeon's intent, and unresponsive robotic systems. Approximately 10-15% of surgical patients in the UK experience adverse events, of which 50% are preventable [6], while 10,624 adverse events in robotic procedures were reported in the US from 2000 to 2013 [7]. Technical errors during surgery have become a leading cause of postoperative complications, resulting in reoperations and readmissions [8]. A lack of standardized RMIS training is identified as one of the main reasons for intraoperative risk to patients [9].

In this context, real-time surgical safety feedback is crucial in mitigating the risks associated with RMIS [10], [11]. While [12] focuses on anomaly detection, we emphasize the importance of error detection as a key component. By providing immediate feedback to surgeons during live surgeries, real-time error detection mechanisms can alert surgeons about potential adverse events, and allow for immediate remedy actions to avoid complications. In surgical training and education, real-time error detection can assist trainees in immediately recognizing and correcting their mistakes by pinpointing areas for improvement, thereby accelerating the learning curve and education efficacy [13]. Furthermore, error detection contributes to more detailed surgical skill assessment. According to [14], fluctuations in the Global Rating Scale during surgery indicate suboptimal performance, with significant deviations suggesting the occurrence of human errors. Once identified, these can serve as valuable indicators of surgical proficiency [15].

However, real-time error detection poses significant challenges due to the complicated nature of surgical procedures and the human involvement in operating surgical robots. For instance, although repeated attempts are regarded as errors, certain actions may be repeated intentionally by the surgeon to achieve the desired outcome. Hutchinson et al. [16] have conceptualized the surgical operation as a hierarchical structure, ranging from the overall procedure down to the specific gesture

Manuscript received: July, 6, 2024; Revised September, 21, 2024; Accepted October, 19, 2024.

This paper was recommended for publication by Editor Pietro Valdastri upon evaluation of the Associate Editor and Reviewers' comments. This work was supported in whole, or in part, by Ministry of Education Tier 1 Start up grant, NUS, Singapore (A-8001267-01-00); Ministry of Education Tier 1 grant, NUS, Singapore (A-8001946-00-00); the Wellcome/EPSRC Centre for Interventional and Surgical Sciences (WEISS) [203145/Z/16/Z, NS/A000050/1]; the EPSRC-funded UCL Centre for Doctoral Training in Intelligent, Integrated Imaging in Healthcare (i4health) [EP/S021930/1]; a UCL Research Excellence Scholarship; the Department of Science, Innovation and Technology (DSIT); the Royal Academy of Engineering under the Chair in Emerging Technologies programme. For the purpose of open access, the authors have applied a CC BY public copyright licence to any author accepted manuscript version arising from this submission.

[1] Zhimin Shao is with the Department of Electronic Engineering, Tsinghua University, Beijing, China. She did this work during her internship at National University of Singapore shaozhimin00@126.com

[2] Jialang Xu, Danail Stoyanov, and Evangelos Mazomenos are with the Wellcome/EPSRC Centre for Interventional and Surgical Sciences and the Department of Medical Physics and Biomedical Engineering, University College London, London, UK {jialang.xu.22; danail.stoyanov; e.mazomenos}@ucl.ac.uk

[3] Yueming Jin is with the Department of Biomedical Engineering and the Department of Electrical and Computer Engineering, National University of Singapore, Singapore ymjin@nus.edu.sg

Zhimin Shao and Jialang Xu contributed equally. Joint senior and corresponding authors: Yueming Jin and Evangelos Mazomenos.

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

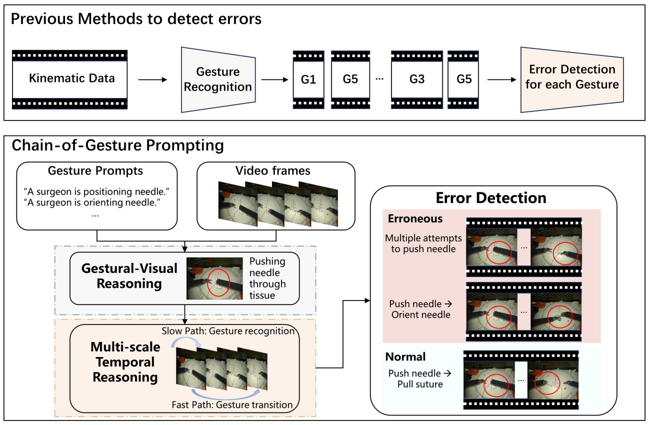
Previous Methods to detect errors
Kinematic Data
Gesture Recognition
G1
G5
...
G3
G5
Error Detection for each Gesture
Chain-of-Gesture Prompting
Gesture Prompts
"A surgeon is positioning needle."
"A surgeon is orienting needle."
...
Video frames
Gestural-Visual Reasoning
Pushing needle through tissue
Multi-scale Temporal Reasoning
Slow Path: Gesture recognition
Fast Path: Gesture transition
Error Detection
Erroneous
Multiple attempts to push needle
Push needle → Orient needle
Normal
Push needle → Pull suture

Fig. 1. Illustration on previous methods and our proposed Chain-of-Gesture prompting. (a) Previous methods detect errors with two separate parts: gesture recognition and error detection for each type of gesture. (b) We propose an end-to-end Chain-of-Gesture prompting framework to capture complex visual reasoning processes with two reasoning modules: Gestural-Visual reasoning and Multi-scale Temporal Reasoning.

and motion. They have annotated the open-source JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS) [17] with error labels through human inspection of videos and the available gesture labels and also introduced a framework for assessing both executional and procedural mistakes by analyzing kinematic data. The findings illustrate that error types and frequencies significantly differ in various tasks and gestures (e.g., pulling a suture, or passing needles).

Recent advances in surgical error detection have led to the development of two separate parts: gesture recognition and error detection within each gesture type [18]. Early works utilized conventional deep learning techniques, e.g., convolutional neural networks (CNNs) and long short-term memory (LSTM), to predict potential unsafe events caused by unintentional human errors in simulated surgical training tasks [18], [19] and retinal microsurgery [20]. Li et al. [19] designed a Siamese network to contrast the trajectories of normal and erroneous gestures and improved the error detection for each type of gesture to an online mode, reporting state-of-the-art performance on JIGSAWS. However, most of these methods rely heavily on the gesture label as prior knowledge to segment the surgical video. The overall performance of error detection in the two-part framework depends on human expertise to annotate gesture labels for each gesture instance in advance, or on the results from gesture recognition that initially segments and categorizes a surgical video into gesture clips. This motivates us to explore the end-to-end error detection method to avoid gesture annotation.

Another limitation of prior work is ignoring contextual and semantic information in surgery, since they focus on analyzing fragments of kinematic data instead of the whole surgical video [21]–[23]. Although kinematic data is informative, it captures only the dynamics of surgical tools. Surgical errors are also expected to be influenced by factors like the surgical tools employed and their interactions with anatomical structures within the surgical workspace [24], aspects that the endoscopic video can capture directly. Besides, the error duration has high variation and errors occur in different time scales. Error types like multiple attempts usually occur across actions and therefore last a long time while the error out of view occurs within the gesture, thus lasting short. This semantic information is also contained in videos and not present in kinematic data [25]. While video-based methods have been extensively validated to achieve comparable or even superior performance to kinematic-based methods in other tasks within surgical data science, such as skill assessment [14], [26], [27], gesture recognition [25] and instrument segmentation [28], the effectiveness of video-based approaches that capture short and long term temporal information in error detection remains unexplored.

In this paper, we propose a novel Chain-of-Gesture (COG) prompting framework for real-time surgical error detection from robotic surgical videos. Inspired by chain-of-thought [29] prompting used in natural language processing that divides a problem into a sequence of intermediate reasoning steps, our COG model consists of sequential reasoning steps to streamline the error detection process. Our approach involves two reasoning modules that mimic the two parts of gesture recognition and error detection within each gesture type, ultimately forming an integrated end-to-end error detection framework, as shown in Fig. 1. Concretely, we first propose Gestural-Visual Reasoning (GVR), which uses language prompts with attention, to locate gestures in videos. We use vision-language models to generate language prompts based on a vocabulary of gestures predefined by experts based on common practices and standard definitions, and then augment the video with additional informative gesture cues without extra annotation cost through a transformer layer and an attention layer. Based on the augmented features, we further develop the Multi-Scale Temporal Reasoning (MSTR) to capture both slow and fast temporal transitions. It is achieved by two temporal feature extraction paths in different time scales and prediction consistency loss across multi-scales. We extensively evaluate our method on the JIGSAWS dataset. Our method outperforms existing state-of-the-art approaches significantly. The main contributions of the paper are as follows:

- We introduce a real-time, end-to-end surgical error detection framework by chain-of-gesture prompting without extra gesture labels.
- Our model incorporates gesture clues with videos through GVR and analyzes surgical procedures at both fine and coarse temporal scales through MSTR with 1.7% and 2.2% improvement in F1 score respectively, enhancing the fine-grained and overall understanding of the surgical context.
- Our model significantly outperforms state-of-the-art methods in surgical error detection.

II. Methods

An overview of our proposed COG is shown in Fig. 2. We begin by formulating the surgical error detection problem, then describe the GVR and MSTR modules of COG. Finally, we use prediction consistency loss across multiple scales to enhance accuracy and consistency in predictions.

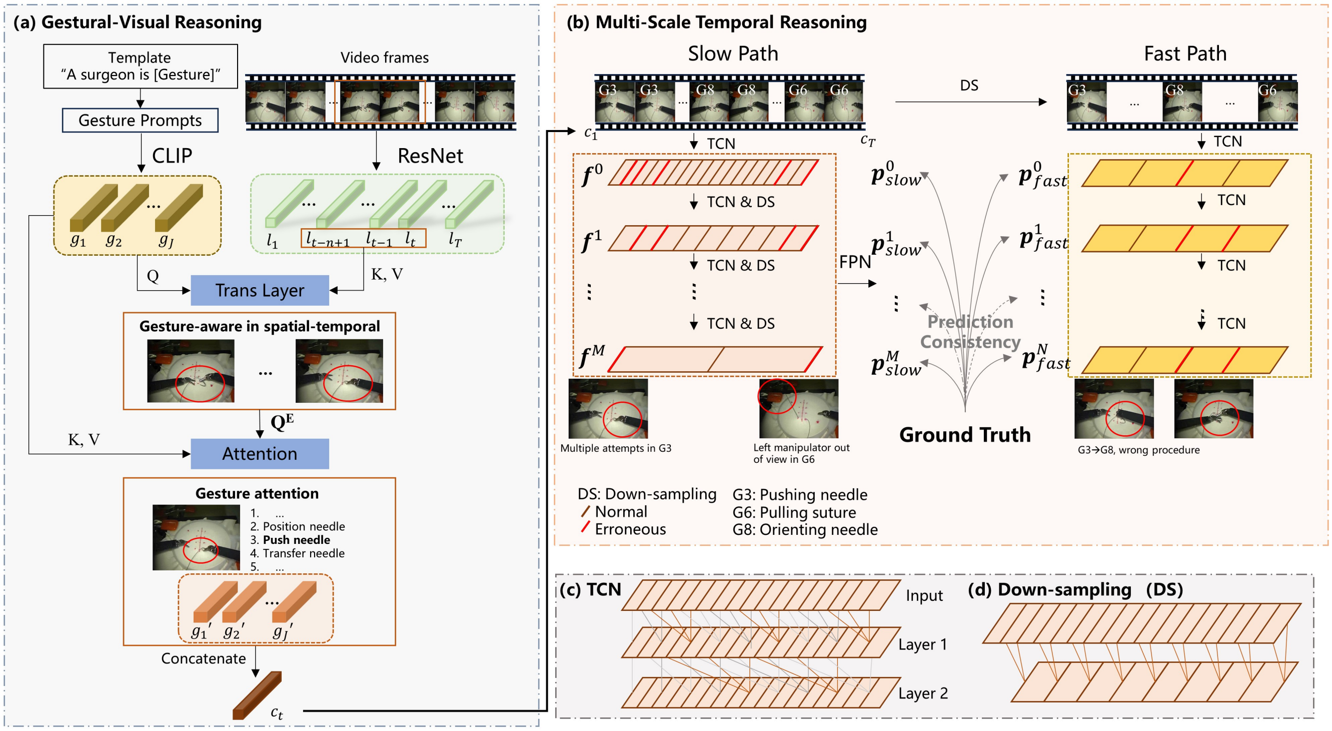

Fig. 2. Overview of our proposed Chain-of-Gesture. (a) Gestural-visual reasoning module with gesture prompts and visual embedding, a transformer layer, and an attention layer for gestural prompting. (b) Multi-scale temporal reasoning module with a slow path and a fast path is optimized by prediction consistency. (c) Temporal Convolutional Network (TCN) in detail. (d) Downsampling in detail.

A. Problem Formulation

During surgical procedures, both human surgeons and robotic systems can make errors that impact outcomes. Surgical errors are generally categorized into executional and procedural errors [16]. Executional errors include repeated attempts of an action or misplacement of instruments beyond the endoscopic view, while procedural errors involve omissions or incorrect sequencing of gestures that are correctly performed in isolation. Procedural errors can be easily detected using grammar graphs [16], so our focus is primarily on executional error detection. Unlike prior studies [18], [19] that analyze kinematic data within predefined temporal windows to detect executional errors for every type of gesture, our study aims to identify errors, in real-time by analyzing surgical videos without relying on gesture labels.

Given a surgical video dataset $(\mathcal{X}, \mathcal{Y})$, where $\mathcal{X}$ represents video frames and $\mathcal{Y}$ contains error labels categorizing each frame as erroneous or normal, our model processes a continuous video stream $X_t = \{x_i\}_{i=1}^{t}$ to identify errors y_t in the current frame x_t in real-time. Here, x_i refers to the i-th video frame in the sequence X_t.

B. Gestural-Visual Reasoning (GVR)

Surgical procedures can be viewed as hierarchical structures composed of a sequence of tasks, each involving discrete steps known as gestures [16]. These gestures represent fundamental surgical actions and error frequency and types vary significantly across different gestures. For instance, pushing needle through tissue often leads to multiple attempt errors due to incorrect angle or depth, while pulling sutures may cause out-of-view errors. Therefore, modeling gesture dynamics within endoscopic videos is essential for understanding the semantic aspects of surgical scenes.

To achieve this goal, we propose a GVR module by structuring the visual features of video frames around a predefined set of gesture prompt features, as shown in Fig. 2 (a). Firstly, we employ a gesture prompt template $t(\cdot)$: "`A surgeon is [Gesture]`", where `[Gesture]` is one of J predefined gestures. Subsequently, we generate gesture prompt feature set $\{g_j\}_{j=1}^{J} \in R^{J \times d_{text}}$ utilizing the CLIP text encoder [30], where d_{text} is the dimension of textual embedding. Thanks to CLIP's strong generalization capabilities, fine-tuning is not required.

$$g_j = CLIP(t(\text{Gesture}_j)), j \in [1, J] \tag{1}$$

For the analysis of spatial information, the current video frame x_t is forwarded to a standard CNN model (ResNet50 [31] in our work due to its validated effectiveness in surgical field [14], [32]) to extract discriminative spatial embeddings $l_t \in R^{2048}$. Subsequently, we enhance the gesture prompt features with spatial awareness by introducing a Transformer layer coupled with an Attention layer [33]. The transformer layer identifies which parts of frames are most similar to the current gesture prompt by scanning through the last n frames and measuring the similarity between gesture prompts and spatial embeddings derived from video frames. Then, the attention layer complements the transformer by focusing specifically on identifying all n recent video frames to the most relevant gesture prompt to obtain spatial-aware gesture prompt features.

Specifically, the transformer layer employs a multi-head attention mechanism, where the gesture prompt feature set $\{g_j\}_{j=1}^{J}$ act as the query, and the sequence of visual features $l_{t-n+1:t}$ with length n serves as both key and value, which is sequentially followed by a layer normalization, a feed-forward layer, and another layer normalization [33], outputting the refined gesture prompt features $Q^E \in R^{J\times d}$, integrating both gestural and visual cues from the recent video frames:

$$Q^E = \text{Trans}\left(\{g_j\}_{j=1}^{J}, l_{t-n+1:t}, l_{t-n+1:t}\right) \tag{2}$$

The attention layer utilizes the scaled dot-product attention mechanism:

$$\text{Atten}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d}})V \tag{3}$$

where $Q, K, V, \sqrt{d}$ denote query, key, value, and scaling factor respectively. The attention layer takes Q^E as the query and the original gesture prompt feature set $\{g_j\}_{j=1}^{J}$ as both the key and value to produce spatial-aware gesture prompt features $g'_j \in R^d$. This mechanism allows for precisely adjusting the attention weights, ensuring that the model focuses on the most relevant spatial-aware features for each gesture prompt.

$$g'_j = \text{Atten}\left(Q^E, \{g_j\}_{j=1}^{J}, \{g_j\}_{j=1}^{J}\right), \quad j \in [1, J] \tag{4}$$

For comprehensive spatial-aware gesture representation, we concatenate all J spatial-aware gesture prompt features, generating a cohesive feature $c_t \in R^{Jd}$ for the current frame x_t. This aggregation enhances the model's ability to interpret and reason about the spatial dynamics of each gesture within the surgical video.

C. Multi-Scale Temporal Reasoning (MSTR)

Out of view error typically occurs within fine temporal scales, whereas multiple attempts happen over broader temporal scales. Drawing inspiration from the SlowFast architecture for video recognition [34], we introduce an MSTR module to address the complexities of temporal dynamics in surgical video analysis, as shown in Fig.2 (b). MSTR is bifurcated into two distinct pathways: a Slow Path and a Fast Path to model temporal information at the granular level of individual frames and identify transitions between gestures at the segment level separately. Unlike the SlowFast model [34], which reduces the entire video to just two frames, our Fast Path down-samples frame-level cohesive features $c_{1:t}$ by average pooling across every 16 frames to identify transitions between gestures at the segment level, and our Slow Path, which processes the frame-level cohesive features $c_{1:t}$ and coarsens as the stages go deeper.

Our model encodes temporal cues for both the Slow and Fast Paths by employing a Temporal Convolutional Network (TCN) [34]. Specifically, the Fast Path incorporates a Multi-Stage TCN (MS-TCN) comprising an initial stage of 11 stacked residual dilated causal 1D convolution layers to generate prediction $\boldsymbol{p}^0_{\text{fast}}$ from the initial stage, followed by N refinement stages, each with 10 causal dilated 1D convolution layers, to obtain refined predictions $\{\boldsymbol{p}^i_{\text{fast}}\}_{i=1}^{N}$. The configuration of TCN is illustrated in Fig. 2 (c) as an example of a 2-layer TCN. For each layer in TCN, the operation can be formulated by

$$\begin{aligned} Z_l &= \text{ReLU}(W_{1,l} * F_{l-1} + b_{1,l}) \\ F_l &= F_{l-1} + W_{2,l} * Z_l + b_{2,l} \end{aligned} \tag{5}$$

where F_l is the output of the layer l, $*$ denotes the convolution operator, $W_{1,l}$ is the causal dilated 1D convolution kernel [35], $W_{2,l}$ is the weight of a 1D convolution and $b_{1,l}$, $b_{2,l}$ are bias vectors.

The Slow Path utilizes a similar MS-TCN architecture but differs by applying MS-TCN to features directly, rather than predictions. Firstly, we employ a TCN to generate the initial feature $\boldsymbol{f}^0$ as the first stage. As the network delves into deeper stages, this path systematically down-samples the temporal resolution by using average pooling with both kernel size and stride set to k (shown in Fig. 2 (d)) at each stage, aiming to compress temporal information effectively. Subsequently, a Feature Pyramid Network (FPN) [36] aggregates features of varying scales $\{\boldsymbol{f}^i\}_{i=0}^{M}$ to synthesize multiple predictions $\{\boldsymbol{p}^i_{slow}\}_{i=0}^{M}$. We take $\boldsymbol{p}^0_{slow}$ as the final frame-level prediction.

$$\boldsymbol{f}^0 = \text{TCN}(c_{1:t}) \tag{6}$$

$$\boldsymbol{f}^i = \text{AvgPool}\left(\text{TCN}(\boldsymbol{f}^{i-1})\right) \tag{7}$$

$$\{\boldsymbol{p}^i_{slow}\}_{i=0}^{M} = \text{FPN}(\{\boldsymbol{f}^i\}_{i=0}^{M}) \tag{8}$$

where $\text{TCN}(\cdot)$ means a single-stage TCN.

D. Prediction Consistency across Multi Scales

For a video comprising a total of T frames, the prediction length produced by Fast Path becomes $\lfloor T/16 \rfloor$ after down-sampling. In Slow Path, the temporal length of predictions at each stage is determined by $T^{i+1} = \lfloor T^i/k \rfloor$, where T^i represents the temporal length of the i-th stage. Therefore, predictions from different stages are in various time scales. SlowFast [34] and SF-TMN [37] combine the predictions from slow and fast paths to generate final predictions to merge coarse and fine temporal information. However, the effectiveness is hindered by constraints in prediction accuracy. To avoid temporal misalignment, inspired by SAHC [38], we ensure prediction consistency by adapting the ground truth y_t to align with the temporal resolution of each stage in both the Slow Path and Fast Path through down-sampling. Subsequently, the losses across all stages in two paths are aggregated to compute the total loss. Within each stage, the loss is composed of two parts: a Cross-Entropy (CE) loss calculated at each time point to assess the accuracy of predictions, and a Mean Squared Error (MSE) calculated over the detection probabilities between every two adjacent time points to ensure smoothness in the prediction sequence.

$$\begin{aligned} \mathcal{L}_{CE} &= \mathcal{L}_{CE-slow} + \mathcal{L}_{CE-fast} \\ &= -\frac{1}{M+1}\frac{1}{T^i}\sum_{i=0}^{M}\sum_{t=1}^{T^i} y_t^i \log(p^i_{slow,t}) \\ &\quad - \frac{1}{N+1}\frac{1}{\lfloor T/16 \rfloor}\sum_{j=0}^{N}\sum_{t=1}^{\lfloor T/16 \rfloor} y_t^j \log(p^j_{fast,t}) \end{aligned} \tag{9}$$

$$\begin{aligned} \mathcal{L}_{MSE} &= \mathcal{L}_{MSE-slow} + \mathcal{L}_{MSE-fast} \\ &= \frac{1}{M+1}\frac{1}{T^i}\sum_{i=0}^{M}\sum_{t=1}^{T^i} |p^i_{slow,t} - p^i_{slow,t-1}|^2 + \\ &\quad \frac{1}{N+1}\frac{1}{\lfloor T/16 \rfloor}\sum_{j=0}^{N}\sum_{t=1}^{\lfloor T/16 \rfloor} |p^j_{fast,t} - p^j_{fast,t-1}|^2 \end{aligned} \tag{10}$$

where y_t^i and y_t^j are the corresponding ground truth at i-th stage in Slow Path and j-th stage in Fast Path, respectively.

Hence, the overall objective of our COG is

$$\mathcal{L}_{total} = \mathcal{L}_{CE} + \lambda \mathcal{L}_{MSE} \tag{11}$$

III. Experiments

A. Datasets and Evaluation Metrics

JIGSAWS [17] is a public dataset derived from the *da Vinci* surgical system, capturing data from eight surgeons performing three dry lab surgical tasks: suturing, knot tying, and needle passing. The dataset includes synchronized kinematic and 640×480 resolution video data, recorded at 30Hz. Hutchinson et al [16] extends JIGSAWS by annotating executional errors at the frame level for suturing and needle-passing tasks. In our research, we utilize the video data and all corresponding error labels, with each video frame being categorized as either normal (0) or erroneous (1). Note that we only use the description of potential gestures from the gesture vocabulary provided by [17], rather than the detailed gesture ground truth of each frame. Following previous work on surgical error detection [19], we employ the Leave-One-Supertrial-Out (LOSO) cross-validation [17] to evaluate our method. All surgeons repeated each surgical task five times. In LOSO, the i-th trial of each surgeon is excluded from the dataset to serve as the test set, thereby assessing the model's ability to generalize across different trials conducted by the same surgeon.

This work aims to detect errors in surgical video in real-time. Therefore, we evaluate the performance of our combined data from the Suturing and Needle Passing tasks using metrics such as the binary F1 score, accuracy, and Jaccard index at the frame level. Furthermore, to ensure an explicit and fair comparison with the state-of-the-art work on surgical error detection [19], we follow its evaluation protocol to generate window-level metrics. Specifically, we apply a 2-second sliding window with a 1.2-second stride to the frame-level predictive labels. Within each window, we average the predictions and binarize them using a threshold of 0.5 to generate the window-level predictive labels. The ground truth in the test set for each trial out under the LOSO settings is detailed in Table I.

TABLE I
The number of frames and windows for each trial in LOSO.

	#frames		#windows	
LOSO settings	Total	Erroneous	Total	Erroneous
Trial 1 out	8332	5453 (65%)	1076	764 (71%)
Trial 2 out	6056	3486 (58%)	775	466 (60%)
Trial 3 out	7066	3739 (53%)	886	500 (56%)
Trial 4 out	6979	3277 (47%)	868	448 (52%)
Trial 5 out	5433	2353 (43%)	640	308 (48%)

B. Implementation Details

All experiments are implemented in PyTorch on a single NVIDIA RTX 3060 GPU. For the extraction of spatial embeddings l_t, we employed the ResNet-50 model, initially pre-trained on the ImageNet dataset and further fine-tuned on the JIGSAWS dataset with error labels frame-by-frame using Adam optimizer with the learning rate of 1×10^{-4} and a batch size of 64. Video frames are resized into 240×240 and center-cropped to 224×224. To reduce redundancy and computational demands, video data are downsampled to 5 Hz. For gesture prompt feature g_j, we used the pre-trained CLIP ViT-B32 [30] model with fixed parameters as our text encoder. The description of gestures is drawn from a common gestural vocabulary [17] comprised of 15 distinct gestures, thus $J = 15$. The spatial and gestural embeddings extracted from ResNet-50 and CLIP are used as inputs to our COG model, without further tuning during the COG model's training phase. Our COG is trained end-to-end using the Adam optimizer for 50 epochs with the initial learning rate set to 5×10^{-4}. We set $M = N = 3$ empirically and standardize the dimension of all casual dilated 1D convolution layers in the MSTR module to 64. The coefficient of MSE loss λ is empirically set to 0.15. The length of sequence n is set to 40, and the kernel size and stride of average pooling k in down-sampling is set to 4. It takes approximately 3 hours for every 50 epochs, where fine-tuning ResNet-50 requires about 2.5 hours and training COG takes approximately 40 minutes for each trial out.

C. Comparison with State-of-the-Art

The existing studies that use CNN or LSTM based on kinematic data for surgical error detection [18], [19], report state-of-the-art F1 scores on JIGSAWS. To ensure a fair comparison, we re-implemented this kinematic-based Siamese-LSTM approach and a range of state-of-the-art video-based methods for surgical video analysis as baselines. These methods include ResNet-50 [31]; TeCNO [39], which employs MS-TCN to capture long-range dependencies in video sequences; Trans-SVNet [32], which introduces a transformer layer to integrate spatial and temporal features effectively; SAHC [38], which explores the use of hierarchical clustering to refine the feature extraction process; and SF-TMN [37], which combines features from slow and fast processing paths to enhance motion analysis. All these methods are classification models that can be directly extended to error detection. We implemented them using their publicly released codes and trained and evaluated them on JIGSAWS dataset under LOSO settings. The comparative results are presented at both the frame-level and window-level in Table II.

Comparing different inputs to the model, video data consistently outperforms kinematic data in error detection. Notably, even the spatial information extracted through a fine-tuned ResNet-50 enhances performance significantly, with improvements observed in the window-level metrics: approximately 2.8% in F1 score, 2.6% in accuracy, and 3.5% in Jaccard, further showcasing the importance of spatial information in the accurate detection of errors.

Among the methods that emphasize temporal information extraction (i.e. TeCNO, Trans-SVNet, SAHC, and SF-TMN), Trans-SVNet emerges as a strong contender. It effectively combines spatial and temporal data using a Transformer-based fusion head, making it the second-best method with an impressive 74.2% F1 score and 59.3% Jaccard. This achievement indicates that contextual information fusion is effective for the surgical error detection task. Nevertheless, it is crucial to note

TABLE II

QUANTITATIVE RESULTS ON FRAME LEVEL AND WINDOW LEVEL OF COMPARISONS BETWEEN SOTA METHODS AND OUR PROPOSED CHAIN-OF-GESTURE MODEL ON JIGSAWS DATASET. Δ GVR: NO GESTURAL-VISUAL REASONING; Δ MSTR: NO MULTI-SCALE TEMPORAL REASONING; Δ SLOW PATH: NO SLOW PATH MODULE; Δ FAST PATH: NO FAST PATH MODULE. **K**: KINEMATIC DATA, **V**: VIDEO DATA. * DENOTES METHODS FOCUSING ON TEMPORAL INFORMATION EXTRACTION.

Input	Method	Frame level			Window level				Inference rate
		F1	Accuracy	Jaccard	F1	Accuracy	Jaccard	P Values	(ms per frame)
K	Siamese-LSTM G*T* [19]	–	–	–	70.0 ± 1.8	65.2 ± 1.2	53.9 ± 2.2	**2e-12**	–
V	ResNet	70.8 ± 4.0	66.9 ± 3.3	54.9 ± 4.9	72.8 ± 4.1	67.8 ± 3.1	57.4 ± 5.3	**5e-09**	6.05
	TeCNO* [39]	69.6 ± 2.6	66.4 ± 2.4	53.4 ± 3.1	71.4 ± 2.3	66.7 ± 2.4	55.5 ± 2.8	**6e-12**	6.09
	Trans-SVNet* [32]	71.0 ± 5.5	59.3 ± 9.0	55.3 ± 6.9	74.2 ± 5.3	62.6 ± 8.7	59.3 ± 6.9	**4e-02**	6.39
	SAHC* [38]	70.7 ± 3.5	67.8 ± 2.2	54.8 ± 4.3	72.6 ± 3.1	68.0 ± 2.2	57.0 ± 4.0	**1e-20**	6.43
	SF-TMN* [37]	69.8 ± 3.4	66.9 ± 2.5	53.7 ± 4.1	71.4 ± 3.4	67.1 ± 2.7	55.7 ± 4.3	**3e-11**	6.86
	Ours (Δ GVR)	70.8 ± 4.2	67.0 ± 3.1	54.9 ± 5.2	72.9 ± 4.4	67.9 ± 3.6	57.5 ± 5.7	**3e-24**	–
	Ours (Δ MSTR)	70.0 ± 4.6	64.8 ± 4.3	54.1 ± 5.7	72.4 ± 4.9	66.3 ± 4.7	57.0 ± 6.3	**3e-11**	–
	Ours (Δ Slow Path)	71.3 ± 4.8	64.0 ± 4.3	55.6 ± 6.0	74.0 ± 4.9	66.2 ± 5.0	59.0 ± 6.5	**1e-02**	–
	Ours (Δ Fast Path)	71.2 ± 3.9	66.4 ± 4.2	55.4 ± 4.9	73.6 ± 4.3	67.9 ± 4.5	58.4 ± 5.6	**7e-03**	–
	Ours	**72.3 ± 4.6**	**68.3 ± 3.5**	**56.8 ± 6.0**	**74.6 ± 5.1**	**69.8 ± 4.5**	**59.8 ± 6.8**	–	6.69

that the accuracy of Trans-SVNet is considerably lower. This suggests a tendency towards generating false positives which could be attributed to the imbalanced dataset. Other methods, despite their sophistication in temporal information extraction, fall short when compared to the more straightforward ResNet-50 approach, possibly due to a misalignment in the temporal scale they employ. While these methods are adept at phase recognition within surgical videos—where the temporal scope is broader and encompasses entire surgical procedures—they appear less suited for the more granular task of error detection. Error detection requires a finer temporal resolution to capture subtle deviations or incorrect actions. Hence, the temporal granularity and feature selection become critical factors in the performance of these models.

Our COG model enhances data with gesture prompts for each frame and extracts multi-scale features at fine and coarse temporal resolutions, resulting in superior performance across all metrics and achieving an F1 score of 72.3% at the frame level (79.3% for Suturing and 60.8% for Needle Passing). Compared to Trans-SVNet, our method achieves similar F1 scores (74.6%) while significantly improving accuracy with lower false positive and false negative rates. Reducing false positives minimizes unnecessary alerts, allowing surgeons to focus on critical tasks, while lowering false negatives enhances patient safety by ensuring timely error detection. This not only highlights higher performance metrics but also represents a methodological advancement in managing gestural and contextual information, emphasizing the potential for further improvements in surgical error detection.

In our statistical analysis, we conducted a paired T-test between COG and other methods to calculate P values for the F1 score at the window level. COG shows a significant improvement in F1, with P values well below 0.05 in all comparisons, confirming its robustness and reliability in detecting errors in surgical videos. We have also evaluated under LOUO settings and the results are presented in the supplementary video Table R1. Our model achieves the highest F1 score, Jaccard index and comparable accuracy with SOTA method.

D. Ablation Studies

1) Effectiveness of Key Components: We first analyze the contributions of two distinct reasoning modules (i.e., GVR and MSTR) in our proposed COG model and evaluate the discrete effects of the "Slow Path" and "Fast Path" in the MSTR, as presented in Table II. For this analysis, we directly use the extracted visual features $l_{1:T}$ as input to the MSTR to obtain ΔGVR, and we pass the cohesive feature c_t through a simple linear head to obtain ΔMSTR setting. The omission of either GVR or MSTR led to performance degradation, as evidenced by all metrics. Through statistical analysis, we determined the P values, which quantified the significance of the performance drop. Notably, the absence of GVR had a more profound impact than the exclusion of MSTR, with a smaller P value. This result confirms our hypothesis that the integration of gesture cues via GVR is not merely beneficial but vital to the model's success, as it provides critical contextual information for reasoning about surgical actions. As for the temporal dynamics of MSTR, our findings indicate that the Fast Path plays a more crucial role than the Slow Path in our context of error detection to reflect the gesture transition.

2) Length of Sequence in GVR: We then focus on the critical parameter of sequence length, denoted as n, which determines the number of video frames considered for identifying its gesture. Figure 3 shows the model performance with different values of n used in Eq. 2. The empirical findings reveal that a sequence length of 40 frames yields the most promising results. This length corresponds to approximately 1.5 times the average gesture duration in our dataset, which is 27 frames. By extending beyond the average single gesture length, the GVR module is afforded a more holistic view, encompassing not just the gesture itself but also the critical transition phase to the subsequent gesture. This broader temporal window includes the tail end of one gesture and the onset of the next, thus we can discern and identify the gesture and its executional errors.

3) Number of Stages in MSTR: The number of stages in MSTR determines the temporal dimensions of the predictions from the final stage. Table III presents the model performance across different values of M and N. We observe that three stages yield the best results; both fewer and greater numbers of stages negatively impact performance. Specifically, when $M = N = 2$, the temporal predictions from the slow path are calculated as $\lfloor T/k^2 \rfloor$. In our experiments, we set $k = 4$. Consequently, the temporal predictions from the final stage of both the slow and fast paths are the same, failing to capture longer ranges of information. As for M=N=4, considering the average video length of 538, a four-fold down-sampling with $k = 4$ reduces the average length of the prediction from the final stage in the slow path to only 2, insufficient to encompass the full range of gestures in the videos, thus limiting the effectiveness.

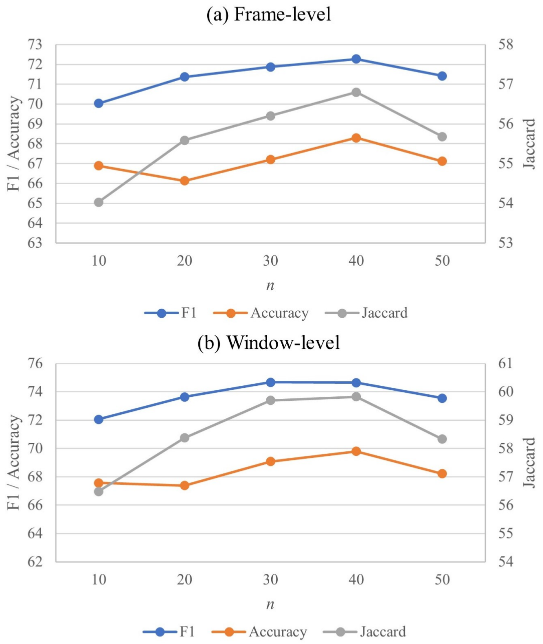

Fig. 3. Analysis of length of sequence n used in GVR. We show the results of the F1 score, Accuracy, and Jaccard of models with different n.

TABLE III
COMPARISON WITH DIFFERENT STAGES M AND N IN MSTR.

M=N	Frame level			Window level		
	F1	Accuracy	Jaccard	F1	Accuracy	Jaccard
2	70.9±4.3	67.3±3.3	55.1±5.3	72.6±4.7	67.9±4.1	57.3±6.1
3	**72.3±4.6**	**68.3±3.5**	**56.8±6.0**	**74.6±5.1**	**69.8±4.5**	**59.8±6.8**
4	71.4±4.1	65.5±3.7	55.6±5.1	73.5±3.9	66.7±4.1	58.3±5.1

TABLE IV
COMPARISON OF TEXTUAL GESTURE PROMPT TEMPLATE.

$t(\cdot)$	Frame level			Window level		
	F1	Accuracy	Jaccard	F1	Accuracy	Jaccard
t_1	71.6±3.2	66.7±3.2	55.9±4.1	73.9±3.5	68.2±3.8	58.8±4.6
t_2	71.8±2.9	67.6±2.7	56.1±3.6	74.3±2.7	68.9±2.9	59.2±3.5
t_3	71.9±4.8	64.9±6.2	56.3±6.2	74.3±6.5	67.4±6.8	59.5±7.2
t_4	**72.3±4.6**	**68.3±3.5**	**56.8±6.0**	**74.6±5.1**	**69.8±4.5**	**59.8±6.8**
t_5	71.2±3.8	67.1±2.5	55.4±4.7	73.7±4.0	68.6±3.2	58.5±5.1

4) Different Template $t(\cdot)$ in GVR: We explored how different textual gesture prompts impact model performance using five templates: (1) $t_1(\cdot)$: direct gesture definition "`[Gesture]`"; (2) $t_2(\cdot)$: gesture definition with a learnable token "`[Gesture][learnable token]`"; (3) $t_3(\cdot)$: CLIP text template "`A photo of [Gesture]`"; (4) $t_4(\cdot)$: our context-specific template "`A surgeon is [Gesture]`"; and (5) $t_5(\cdot)$: more complex sentences for each gesture generated by ChatGPT-4o mini (see Table R2 in the supplementary video for detailed prompts). Results are summarized in Table IV. The direct definition provided moderate metrics but lacked context while adding a learnable token improved results slightly. The CLIP-based template showed better F1 and Jaccard scores, but the lowest accuracy, likely due to its generic nature. In contrast, our context-specific template $t_4(\cdot)$ outperformed all others across metrics, highlighting the importance of domain-relevant context for enhancing model performance. Interestingly, augmenting gestures into complete sentences with ChatGPT resulted in a performance drop. This decline likely stems from the introduction of irrelevant information, which diluted the precision needed for surgical gestures and errors.

E. Visual Results

In Fig. 4, we visually represent the error detection outcomes of a suturing video clip. Our model's incorporation of both contextual and temporal information is evident in the consistency and robustness of its predictions. By capturing a wider range of cues over time, the proposed COG model has an enhanced ability to recognize and flag errors that other methods could miss. Furthermore, we visualize typical errors and results to facilitate an intuitive understanding of the COG model's advantages, such as multiple attempts to push the needle through the tissue and the tool manipulator moving out of the camera's view. Additional example results are provided in the supplementary video.

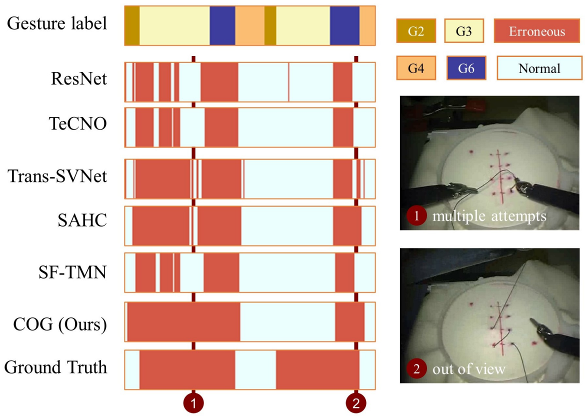

Fig. 4. Color-coded ribbon illustration for a suturing video clip.

IV. DISCUSSION AND CONCLUSION

Our study presents a novel COG prompting approach for error detection in RMIS. Different from traditional spatial-temporal feature extraction from kinematic data, this leverages readily available video data and innovates the error detection paradigm through two reasoning modules that simulate expert human decision-making. By incorporating gestural information, our method effectively discerns executional errors and achieves a notable 4-6% improvement in window-level metrics compared to existing methods on the JIGSAWS dataset, without the need for gesture labels during training–a significant advancement over previous approaches.

The gestural information inherent in surgical videos encompasses a spectrum of behaviors, such as gesture segmentation, recognition, and transition, making it hard to design a one-size-fits-all method for extracting the critical cues. Current methods typically identify gestures and then detect executional errors within the gesture clip sequentially, reliant on the performance of two distinct parts for gesture recognition and error detection within each type of gesture. In contrast, our COG integrates two reasoning modules as an end-to-end framework. The first module focuses on gesture localization and segmentation, while the second dives into details within gestures and transitions, aiming for a comprehensive understanding of the surgical context. The validity of our approach is supported by ablation studies and visual analysis, reinforcing the logic and efficacy of our chain-of-gesture philosophy to think step by step.

Our COG can be applied in clinical practice, thanks to its computational efficiency (6.69 milliseconds/frame). This efficiency is due to our GVR design, which used the transformer architecture known for its parallel processing capabilities and the down-sampling technique in MSTR significantly reduces video length. We have extended our COG method to real surgical videos using the SAR-RARP50 dataset [40]. Compared to Trans-SVNet [32] and SAHC [38], our COG achieves promising performance of 69.98% accuracy, with 1.92% and 4.26% improvement in accuracy, respectively. Our research shows the potential of integrating contextual and temporal data analysis to develop an accurate error detection framework with considerable implications for surgical training. Further work will focus on detecting error types semantically and finding remedial measures, which could significantly enhance the learning curve for novice surgeons.

References

[1] L. Maier-Hein, *et al.*, "Surgical data science for next-generation interventions," *Nat. Biomed. Eng.*, vol. 1, no. 9, pp. 691–696, 2017.

[2] L. Maier-Hein, *et al.*, "Surgical data science–from concepts toward clinical translation," *Med. Image Anal.*, vol. 76, p. 102306, 2022.

[3] C. D'Ettorre, *et al.*, "Accelerating surgical robotics research: A review of 10 years with the da vinci research kit," *IEEE Robot. Automat. Mag.*, vol. 28, no. 4, pp. 56–78, 2021.

[4] P. Joice, *et al.*, "Errors enacted during endoscopic surgery—a human reliability analysis," *Appl. Ergon.*, vol. 29, no. 6, pp. 409–414, 1998.

[5] O. Elhage, *et al.*, "An assessment of the physical impact of complex surgical tasks on surgeon errors and discomfort: a comparison between robot-assisted, laparoscopic and open approaches," *BJU Int.*, vol. 115, no. 2, pp. 274–281, 2015.

[6] C. Vincent, *et al.*, "Adverse events in british hospitals: preliminary retrospective record review," *BMJ*, vol. 322, no. 7285, pp. 517–519, 2001.

[7] H. Alemzadeh, *et al.*, "Adverse events in robotic surgery: a retrospective study of 14 years of fda data," *PloS one*, vol. 11, no. 4, p. e0151470, 2016.

[8] S. E. Regenbogen, *et al.*, "Patterns of technical error among surgical malpractice claims: an analysis of strategies to prevent injury to surgical patients," *Ann. Surg.*, vol. 246, no. 5, pp. 705–711, 2007.

[9] J. W. Collins *et al.*, "Training in robotic surgery, replicating the airline industry. how far have we come?" *World J. Urol.*, vol. 38, pp. 1645–1651, 2020.

[10] K. Liu, *et al.*, "Real-time surgical tool detection in computer-aided surgery based on enhanced feature-fusion convolutional neural network," *J. Comput. Des. Eng.*, vol. 9, no. 3, pp. 1123–1134, 2022.

[11] F. Aspart, *et al.*, "Clipassistnet: bringing real-time safety feedback to operating rooms," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 17, pp. 5–13, 2022.

[12] D. J. Samuel *et al.*, "Unsupervised anomaly detection for a smart autonomous robotic assistant surgeon (saras) using a deep residual autoencoder," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7256–7261, 2021.

[13] H. W. Schreuder, *et al.*, "Training and learning robotic surgery, time for a more structured approach: a systematic review," *BJOG: Int. J. Obstet. Gynaecol.*, vol. 119, no. 2, pp. 137–149, 2012.

[14] D. Anastasiou, *et al.*, "Keep your eye on the best: Contrastive regression transformer for skill assessment in robotic surgery," *IEEE Robot. Autom. Lett.*, vol. 8, no. 3, pp. 1755–1762, 2023.

[15] S. Morita, *et al.*, "Real-time surgical problem detection and instrument tracking in cataract surgery," *J. Clin. Med.*, vol. 9, no. 12, p. 3896, 2020.

[16] K. Hutchinson, *et al.*, "Analysis of executional and procedural errors in dry-lab robotic surgery experiments," *Int. J. Med. Robot.*, vol. 18, no. 3, p. e2375, 2022.

[17] Y. Gao, *et al.*, "Jhu-isi gesture and skill assessment working set (jigsaws): A surgical activity dataset for human motion modeling," in *Med. Image. Comput. Comput. Assist. Interv., M2CAI Workshop*, vol. 3, no. 3, 2014.

[18] M. S. Yasar *et al.*, "Real-time context-aware detection of unsafe events in robot-assisted surgery," in *IEEE/IFIP Int. Conf. Depend. Syst. Netw.*, 2020, pp. 385–397.

[19] Z. Li, *et al.*, "Runtime detection of executional errors in robot-assisted surgery," in *IEEE Int. Conf. Robotics Autom.*, 2022, pp. 3850–3856.

[20] C. He, *et al.*, "Enabling technology for safe robot-assisted retinal surgery: Early warning for unsafe scleral force," in *IEEE Int. Conf. Robotics Autom.*, 2019, pp. 3889–3894.

[21] A. Zia *et al.*, "Automated surgical skill assessment in rmis training," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 13, pp. 731–739, 2018.

[22] H. Ismail Fawaz, *et al.*, "Accurate and interpretable evaluation of surgical skills from kinematic data using fully convolutional neural networks," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 14, pp. 1611–1617, 2019.

[23] B. van Amsterdam, *et al.*, "Gesture recognition in robotic surgery: a review," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 6, 2021.

[24] I. Funke, *et al.*, "Using 3d convolutional neural networks to learn spatiotemporal features for automatic surgical gesture recognition in video," in *Med. Image. Comput. Comput. Assist. Interv.*, 2019, pp. 467–475.

[25] L. Zappella, *et al.*, "Surgical gesture classification from video and kinematic data," *Med. Image Anal.*, vol. 17, no. 7, pp. 732–745, 2013.

[26] I. Funke, *et al.*, "Video-based surgical skill assessment using 3d convolutional neural networks," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 14, pp. 1217–1225, 2019.

[27] T. Wang, *et al.*, "Towards accurate and interpretable surgical skill assessment: A video-based method incorporating recognized surgical gestures and skill levels," in *Med. Image. Comput. Comput. Assist. Interv.*, 2020, pp. 668–678.

[28] Y. Jin, *et al.*, "Incorporating temporal prior from motion flow for instrument segmentation in minimally invasive surgery video," in *Med. Image. Comput. Comput. Assist. Interv.* Springer, 2019, pp. 440–448.

[29] J. Wei, *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," in *Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 24 824–24 837.

[30] A. Radford, *et al.*, "Learning transferable visual models from natural language supervision," in *Int. Conf. Mach. Learn.*, 2021, pp. 8748–8763.

[31] K. He, *et al.*, "Deep residual learning for image recognition," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[32] Y. Jin, *et al.*, "Trans-svnet: hybrid embedding aggregation transformer for surgical workflow analysis," *Int. J. Comput. Assist. Radiol. Surg.*, vol. 17, no. 12, pp. 2193–2202, 2022.

[33] A. Vaswani, *et al.*, "Attention is all you need," in *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[34] C. Feichtenhofer, *et al.*, "Slowfast networks for video recognition," in *IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 6202–6211.

[35] Y. A. Farha *et al.*, "Ms-tcn: Multi-stage temporal convolutional network for action segmentation," in *IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 3575–3584.

[36] T.-Y. Lin, *et al.*, "Feature pyramid networks for object detection," in *IEEE Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 2117–2125.

[37] B. Zhang, *et al.*, "Sf-tmn: Slowfast temporal modeling network for surgical phase recognition," *arXiv preprint arXiv:2306.08859*, 2023.

[38] X. Ding *et al.*, "Exploring segment-level semantics for online phase recognition from surgical videos," *IEEE Trans. Med. Imag.*, vol. 41, no. 11, pp. 3309–3319, 2022.

[39] T. Czempiel, *et al.*, "Tecno: Surgical phase recognition with multi-stage temporal convolutional networks," in *Med. Image. Comput. Comput. Assist. Interv.*, 2020, pp. 343–352.

[40] J. Xu, *et al.*, "Sedmamba: Enhancing selective state space modelling with bottleneck mechanism and fine-to-coarse temporal fusion for efficient error detection in robot-assisted surgery," *arXiv preprint arXiv:2406.15920*, 2024.