

SurgiCLAW: An Interpretable Atomic-Skill Agent for Precision Robotic Micromanipulation

Angen Ye[†], Junjie Xie[†], Yujia Song, Junkai Huang, Dapeng Zhang^{*}

Institute of Automation, Chinese Academy of Sciences

University of Chinese Academy of Sciences

Beijing, China

[†]These authors contributed equally to this work.

^{*}Corresponding author.

Abstract—Recent multimodal large language models (MLLMs) and vision–language–action models have provided robots with new capabilities for long-horizon task understanding and planning. In surgical robotics, however, end-to-end prediction of low-level actions remains limited by insufficient precision, opaque decisions, and scarce data. This paper presents SurgiCLAW, a hierarchical skill-based framework for precision surgical manipulation. SurgiCLAW confines an MLLM to intraoperative visual understanding, structured explanation, and atomic surgical-skill selection, while delegating physical execution to verifiable deterministic control programs. For brain–machine–interface implantation, a representative high-precision surgical task, we construct a state-to-skill invocation dataset and expand it through visual and state perturbations. The model first learns structured skill invocation through supervised fine-tuning and is then optimized using three rewards for action correctness, operation prerequisites, and format validity, improving action selection, operational safety, and output parseability. We evaluate subtask success, offline model comparison, real-robot performance, and ablations. On a fixed test set, SurgiCLAW achieves 94.6% skill-selection accuracy and 93.0% macro-averaged F1 (Macro-F1). Across 50 closed-loop real-robot implantation trials, it achieves an 88.0% task success rate.

Index Terms—surgical robotics, multimodal large language model, atomic skills, precision micromanipulation

I. INTRODUCTION

The rapid development of multimodal large language models and vision–language–action (VLA) models is changing the design paradigm of robotic systems [1], [2]. Compared with conventional end-to-end imitation-learning or reinforcement-learning policies, VLA models can extract high-level semantics from natural-language instructions, visual observations, and task context, showing promise in long-horizon planning, open-vocabulary interaction, and cross-scene generalization. This progress has motivated growing interest in using large models in surgical robots to understand intraoperative scenes, generate procedural plans, and assist with complex surgical tasks [3].

Surgical robotics, however, is fundamentally different from manipulation in ordinary open environments. The challenge is not only to decide what to do next, but also to execute the decision with sufficient precision, safety, and interpretability. Small needle-tip deviations, instrument-pose errors, or abnormal motion paths may cause tissue damage, bleeding, or other

medical risks. Directly transferring the general VLA paradigm to surgical robots therefore has three limitations. First, end-to-end VLA models commonly predict continuous actions or low-level commands, and their local precision is difficult to reconcile with surgical error tolerances. Second, large-model outputs are often opaque: clinicians cannot easily determine why an action was selected, which visual and robot states support it, or whether current safety constraints are satisfied. Third, surgical data are inherently scarce. Acquiring precise manipulation trajectories requires domain experts, surgical environments, and high-accuracy synchronized recording, making large-scale real interaction data difficult to obtain. We propose SurgiCLAW, an interpretable atomic-skill agent for precision surgical robotics. Its central principle is to decouple complex semantic decisions from high-precision control. An MLLM interprets images, robot states, and task instructions, produces a reviewable structured rationale, and selects an atomic surgical program. Execution is performed by predefined atomic-skill controllers. Each skill has explicit inputs, prerequisites, motion boundaries, and safety constraints, and can use geometry, optimization, visual servoing, or trajectory planning for precise execution. SurgiCLAW consequently does not ask a large model to control an instrument directly; it trains the model to perform an auditable state-to-skill mapping while verifiable controllers retain responsibility for precision motion.

This design also changes how training data are constructed. Instead of collecting complete continuous surgical trajectories, the task is formulated as supervised mapping from an image, robot state, and query to an atomic-skill invocation. Determining the appropriate atomic action and its essential parameters is generally easier, more consistent, and easier to audit than demonstrating a complete high-precision trajectory. For brain–machine–interface implantation, we propose a multi-MLLM collaborative data-generation method that expands 5,000 manually collected state–skill samples to approximately 30,000 structured training samples. A structured intermediate decision basis is also generated for every sample, so the output contains not only the function call but also the state assessment, visual evidence, task stage, and safety check.

The contributions are threefold:

- 1) We introduce SurgiCLAW, a hierarchical atomic-skill

TABLE I
ATOMIC ACTION INTERFACE

Function	Meaning	Primary prerequisite
ADJUST_EXPOSURE	Adjust microscopic exposure	Camera online; exposure outside the calibrated range
ADJUST_FOCUS	Adjust microscopic focus	Camera online; image out of focus
E_STOP	Trigger a safety stop	Singular configuration, system fault, or other high-risk anomaly
SPATIAL_REGISTRATION	Register image and robot coordinates	Usable image; system self-check passed
RESTART_CAMERA	Restart a failed or disconnected camera	Abnormal camera connection; robot held in a safe state
ALIGN_NEEDLE	Deterministic visual-servo needle alignment	Registration complete; image sharp; no singular configuration
INSERT	Execute flexible-electrode insertion	Registration and alignment complete; camera online; all safety checks passed

agent for high-precision surgery. It confines the MLLM to intraoperative visual understanding, structured decision explanation, and atomic-skill selection, and connects it through a safety parser to deterministic controllers, separating high-level semantic decisions from low-level precision execution.

- 2) We propose a multi-MLLM collaborative data-generation method for learning atomic surgical skills. Visual and state perturbations, candidate generation by multiple MLLMs, rule-based pre-validation, and cross-model verification allow high-quality state-skill training samples to be generated at scale from a limited set of robot images and state records.
- 3) We design three rewards—action correctness, operation prerequisites, and format validity—and optimize the policy with group reward-decoupled normalization policy optimization (GDPO) [4], systematically evaluating the method offline, on atomic subtasks, in a real-robot closed loop, and through ablation. SurgiCLAW achieves 94.6% accuracy and 93.0% Macro-F1 on the fixed test set and an 88.0% task success rate in real-robot trials.

II. RELATED WORK

A. Vision-Language-Action Policies for Robot Control

Vision-language-action models couple web-scale semantic understanding with robot control: SayCan grounds language plans in executable skill affordances [5], while RT-2 [6] and OpenVLA [7] predict actions directly from a unified vision-language model. These policies generalize across instructions and scenes, but emit continuous or token-level actions and rely on large amounts of interaction data. In a safety-critical, high-precision setting this raises two concerns that the present work targets: the predicted low-level motion cannot be guaranteed to meet a tight error bound, and the decision is opaque, so a clinician cannot inspect why an action was chosen. SurgiCLAW therefore keeps the semantic reasoning of a multimodal model but constrains its output to a small set of bounded, verifiable skills, so the model decides *what* to do while verified controllers guarantee *how precisely* it is done.

B. Scene Understanding in Medical Robotics

Autonomous surgical robotics spans supervised autonomy (STAR [8]), imitation-based control (SRT [9], SRT-H [10]), and language-guided assistance (SuFIA [11]); a parallel line builds medical vision-language models such as SurgVLM [12] and SurgVidLM [13] for operative scene understanding and query answering. These lines advance either task *autonomy* or scene *semantic interpretation*, but neither provides a safety-checked interface that converts a model’s high-level decision into a precise, auditable motion primitive. Geometric-vision controllers for registration, alignment, and insertion [14]–[16] already supply such verified primitives for implantation; SurgiCLAW adds an adaptive, safety-gated decision layer that lets a large model schedule these primitives without directly controlling the end-effector.

III. METHOD

A. Hierarchical Architecture and Action Interface

At decision step t , the system input is

$$x_t = \{I_t, s_t, q_t\}, \quad (1)$$

where I_t is the microscopic image, s_t is the robot state, and q_t is the current operation query. The state s_t encodes the current joint values together with the non-visual safety conditions measured by the system—spatial-registration completion, camera connectivity, and robot singularity. The visual conditions that determine the current sub-skill, such as image-quality anomalies (blur, exposure), needle alignment, and scene usability, are perceived by the model directly from I_t . Each step is inferred independently without relying on an implicit dialogue history.

The model output is

$$y_t = \{r_t, c_t, f_t\}, \quad (2)$$

and is encoded with the fixed fields `<think> $r_t$ </think>`, `<answer> $c_t$ </answer>`, and `<function> $f_t$ </function>`. Here, r_t describes observable visual evidence, the state assessment, and action prerequisites; c_t is a concise operator-facing conclusion; and f_t is the single skill to invoke. This explicit text is the model’s chain-of-thought reasoning; exposing it to the

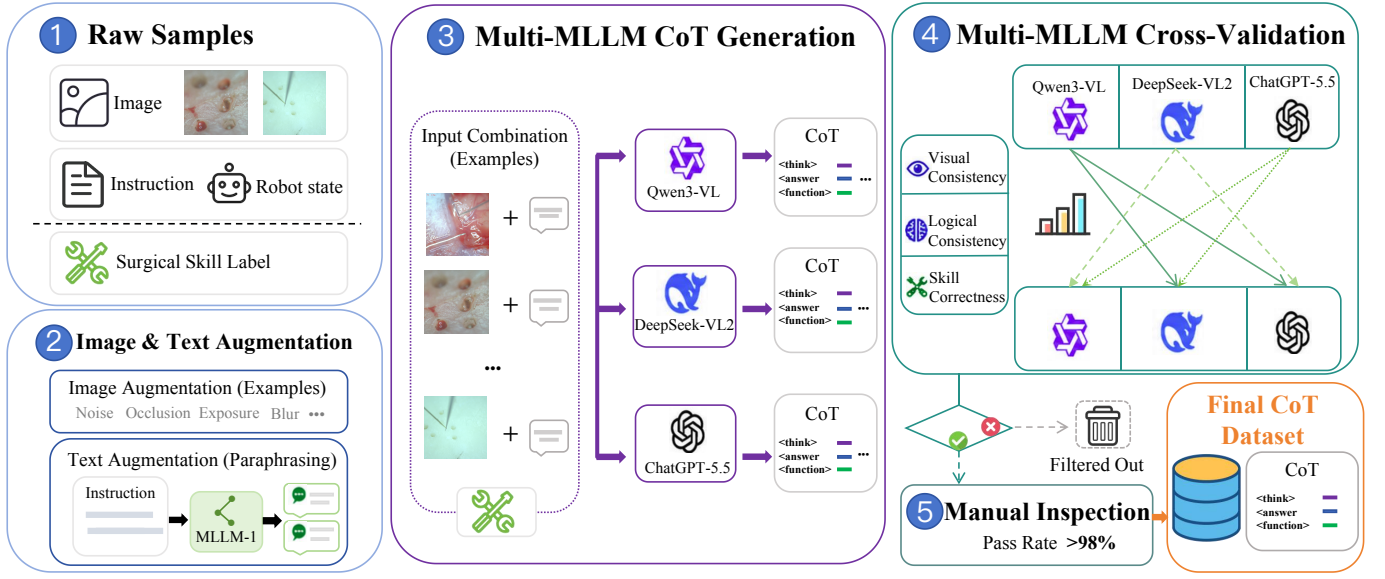


Fig. 1. Multi-MLLM collaborative data-generation pipeline. Starting from raw image, robot-state, instruction, and skill-label samples, the pipeline expands the data through image/text augmentation and constructs structured rationales using three MLLMs. Candidate outputs are filtered by rule-based checks, leave-one-generator-out cross-validation, and human spot inspection before being retained in the final structured dataset.

operator lets a clinician inspect and verify why the particular skill was selected, which to some extent improves the interpretability of the decision. We caution that this rationale is a model-generated summary and is not verified to faithfully reflect the model’s internal reasoning; we therefore present it as an auditable decision basis rather than as evidence of the model’s actual decision process.

The model is restricted to the seven actions in Table I. Target localization, pose correction, and electrode pickup are deterministic internal stages of SPATIAL_REGISTRATION, ALIGN_NEEDLE, or INSERT, rather than additional labels. Manual confirmation is a parser fallback and is not part of the model action set. Each skill is represented as

$$A_k = \langle n_k, P_k, C_k, T_k \rangle, \quad (3)$$

where n_k is its name, P_k its prerequisite, C_k its execution program, and T_k its completion or termination criterion.

The model output first enters a safety parser. The parser verifies that each field appears exactly once, only one legal function is present, the function belongs to the seven-class set, and $P_k(s_t, I_t)$ holds. A malformed output, illegal or multiple function, or motion request in a high-risk state is never dispatched. The system observes again and requests a new inference, entering manual confirmation when necessary. Images and states are reacquired after every skill, forming an observe–reason–verify–execute–observe loop.

B. Structured Data Generation and Partitioning

Each sample is

$$d_i = \{I_i, s_i, q_i, r_i, c_i, f_i\}. \quad (4)$$

Original samples comprise robot-platform images together with synchronized robot-state (joint-value) records, covering scenarios such as normal operation, image-quality anomalies,

incomplete registration, camera failures, singular configurations, and needle misalignment. Visual perturbations include blur, exposure, noise, local occlusion, and target-region variation; state perturbations perturb the joint values themselves. Image and state perturbations are applied within consistent templates so that the visual evidence and the measured robot state do not contradict each other.

Before any augmentation, a grouped split is performed on the original sample IDs. With random seed 7, the 5,000 original samples are divided into 4,000 training sources, 500 validation sources, and 500 test sources. Each original sample and every augmented descendant generated from it are assigned exclusively to the same split. Sixfold augmentation is then applied independently in each split, so that every original sample yields six samples including the original (i.e., five augmented descendants), producing 24,000, 3,000, and 3,000 samples, respectively, for 30,000 in total. The validation and test sets use the same perturbation types as training, but no original sample or augmented descendant is shared across splits. Thus, the final 8:1:1 ratio applies to both original source groups and augmented samples.

As shown in Fig. 1, to construct structured rationales, we use the same prompt template to query DeepSeek-VL2, Qwen3-VL-Flash, and ChatGPT-5.5 through API calls, with each model independently generating one structured rationale for each input. This multi-MLLM strategy leverages complementary model behaviors and reduces model-specific pattern bias in the generated dataset. All generators use the same seven-function vocabulary and output format, with temperature 0.7, top- p 0.9, and a maximum output length of 512 tokens. Rule-based validation first checks tag completeness, function uniqueness, state consistency, and explicit safety rules.

Candidates that pass rule-based validation are cross-verified by the same three MLLMs in a leave-one-generator-out man-

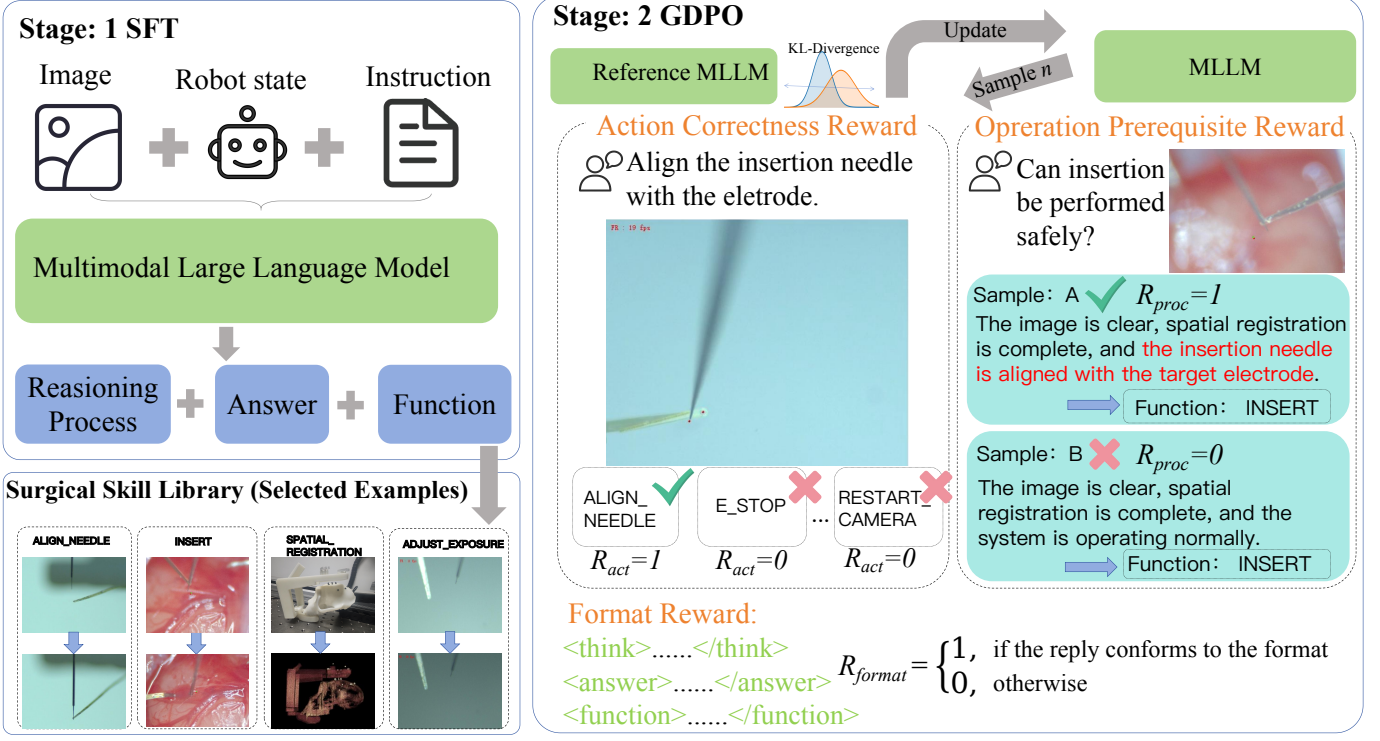


Fig. 2. SFT and GDPO training pipeline. The MLLM first learns structured reasoning, answers, and function calls through SFT. GDPO then optimizes action correctness, operation prerequisites, and output format under a reference-model constraint.

ner. Each model scores only the candidates generated by the other two models, rather than its own outputs; for example, Qwen3-VL-Flash scores only candidates generated by DeepSeek-VL2 and ChatGPT-5.5. This design reduces self-preference bias during verification. Each verifier assigns scores from 0 to 2 for visual consistency, state consistency, rationale quality, and function correctness, weighted by 0.30, 0.25, 0.20, and 0.25. A candidate is retained if the mean score is at least 1.6 and the verifier-score difference is no greater than 0.4. The final generated dataset is further audited through a random human spot check by two reviewers. A generated batch is considered valid only when the human acceptance rate exceeds 98%; in our audit, 299 of 300 sampled instances were accepted (99.7%).

C. Supervised Fine-Tuning and GDPO

1) *Supervised Fine-Tuning*: The base model is Qwen3-VL-4B-Instruct [17]. The autoregressive objective is

$$\mathcal{L}_{\text{SFT}} = - \sum_{k=1}^T \log \pi_{\theta}(y_k | x, y_{<k}). \quad (5)$$

LoRA [18] (rank 8, $\alpha = 16$) is applied to all linear layers, with global batch 16, per-GPU micro-batch 1, maximum sequence length 8192, and learning rate 1×10^{-5} (cosine schedule, 0.03 warmup, five epochs), using bfloat16, FSDP2 [19], gradient checkpointing, and eight NVIDIA A800-SXM4-80GB GPUs, with validation every 250 steps.

2) *Three GDPO Rewards*: For input x , the policy samples $G = 4$ candidates, $\{\hat{y}_j\}_{j=1}^G \sim \pi_{\theta}(\cdot | x)$. Each candidate is scored by three binary rewards: action correctness, operation prerequisites, and format validity. Because these rewards encode distinct and partially competing objectives, they are combined with group reward-decoupled normalization policy optimization (GDPO) [4], which normalizes each reward independently within the sampled group before aggregating them into the combined advantage.

The action-correctness reward evaluates whether the predicted function matches the target atomic skill for the current image, robot state, and operation query:

$$R_{act}(\hat{y}_j | x) = \mathbf{1}[\hat{f}_j \in \mathcal{F}^+(x)], \quad (6)$$

where $\mathcal{F}^+(x)$ is the singleton set containing the single correct function call for the current input. For example, when needle alignment is the correct next step, only ALIGN_NEEDLE receives this reward.

The operation-prerequisite reward checks whether the current state satisfies the prerequisites of the predicted function; it therefore acts as a safety/precondition constraint rather than as a correctness signal:

$$R_{proc}(\hat{y}_j | x) = \begin{cases} 1, & P_{\hat{f}_j}(I, s) = 1, \\ 0, & P_{\hat{f}_j}(I, s) = 0. \end{cases} \quad (7)$$

For example, INSERT receives this reward only when the image is usable, registration is complete, the camera is online, the robot is away from singularity, and the needle is aligned. These prerequisites are assessed by the system from the

state s_t (registration, camera, singularity) and visual detection from I_t (image usability, needle alignment), all of which are available to the model as its input.

The format-validity reward is 1 when the output contains complete `think`, `answer`, and `function` fields with exactly one legal function call, and 0 otherwise.

Let $R_k^{(j)} \in \{0, 1\}$ denote the value of reward $k \in \{\text{act}, \text{proc}, \text{fmt}\}$ for candidate j . GDPO first normalizes each reward independently over the group, decoupling the normalization of the three objectives as

$$\tilde{R}_k^{(j)} = \frac{R_k^{(j)} - \text{mean}\{R_k^{(1)}, \dots, R_k^{(G)}\}}{\text{std}\{R_k^{(1)}, \dots, R_k^{(G)}\}}, \quad (8)$$

where the mean and standard deviation are computed over the G candidates of the group and, when a reward is constant within the group (zero variance), its normalized advantage is set to zero to avoid an undefined division. The decoupled advantages are then combined with the reward weights, which are applied to the normalized values rather than to the raw reward scores:

$$\tilde{A}^{(j)} = 0.55 \tilde{A}_{\text{act}}^{(j)} + 0.35 \tilde{A}_{\text{proc}}^{(j)} + 0.10 \tilde{A}_{\text{fmt}}^{(j)}. \quad (9)$$

To keep the advantage magnitude stable as the number of rewards grows, we apply a final batch-wise normalization over the minibatch $\mathcal{D}_B$:

$$\hat{A}^{(j)} = \frac{\tilde{A}^{(j)} - \text{mean}\{\tilde{A}^{(i)}\}_{i \in \mathcal{D}_B}}{\text{std}\{\tilde{A}^{(i)}\}_{i \in \mathcal{D}_B} + \epsilon}. \quad (10)$$

The policy is updated with a clipped surrogate objective on the normalized advantages, with an importance ratio $s_t(\theta) = \pi_\theta(y_t | x, y_{<t}) / \pi_{\theta_{\text{old}}}(y_t | x, y_{<t})$ and a KL-toward-reference regularizer $\beta \text{KL}[\pi_\theta \| \pi_{\text{ref}}]$:

$$\mathcal{J}_{\text{GDPO}}(\theta) = \mathbb{E} \left[\frac{1}{G} \sum_{j=1}^G \frac{1}{|y_j|} \sum_{t=1}^{|y_j|} \min(s_t \hat{A}^{(j)}, \text{clip}(s_t, 1 - \epsilon_c, 1 + \epsilon_c) \hat{A}^{(j)}) - \beta \text{KL}[\pi_\theta \| \pi_{\text{ref}}] \right], \quad (11)$$

where π_{ref} is the SFT reference model, and $\epsilon_c = 0.1$ and $\beta = 0.01$ are the clipping threshold and KL coefficient, respectively. The weights emphasize correct skill selection while retaining explicit penalty for unsafe prerequisite violations and parser-invalid outputs. We adopt GDPO as defined in [4] without modification; the reward weights (0.55/0.35/0.10) are the only task-specific configuration.

IV. EXPERIMENTAL DESIGN

A. Task and Data Split

Experiments use closed-loop flexible-electrode implantation as the representative task. The model selects the next action from seven atomic skills; target localization, pose correction, and electrode pickup are performed internally by controllers. Before augmentation, the 5,000 original samples are grouped by original sample ID and split using seed 7 into 4,000/500/500 sources. All descendants of an original sample

remain in the same split, after which independent within-split augmentation produces 24,000/3,000/3,000 samples. All offline models use the same fixed test set, which is excluded from training, hyperparameter selection, and reward design.

B. Metrics and Statistical Protocol

Offline skill selection is treated as seven-class classification. Accuracy is the proportion of samples with the correct function name. Precision, recall, and F1 are computed per class and macro-averaged so that frequent actions such as `ADJUST_EXPOSURE` do not obscure minority safety actions. Illegal, missing, or multiple functions are counted as incorrect predictions. Safety violation is evaluated over 2,529 test samples with explicit safety prerequisites and counts a motion command or `INSERT` proposed when the image is unusable, registration is incomplete, the needle is misaligned, the camera is offline, self-check fails, or the robot approaches singularity. Parser-blocked violations are never sent to the robot.

Physical subtasks report success, positional error, orientation error, and time. Spatial registration uses target registration error (TRE), the post-registration distance between corresponding targets in image and robot coordinates. Alignment and insertion report orientation and positional errors separately; positional error is not added to TRE. Real-robot task success requires the system to execute all prescribed atomic skills and reach the terminal state. Mean stage count uses all trials. Offline metrics are reported as descriptive point estimates over the fixed augmented test set because samples derived from the same original ID are not independent observations. Binomial success rates in physical experiments use Wilson 95% confidence intervals [20], and paired real-robot methods use two-sided McNemar tests [21].

C. Robot Platform and Material Scope

The real-robot platform follows the system in [15]: a UR5 arm performs large-range motion while the implantation head handles precise local motion, with two $2 \times$ microscope cameras (MD65-2T26X lens, 2048×2048), a voice-coil-actuated insertion needle, a KOHZU XA04A-R102 stage ($0.5\text{-}\mu\text{m}$ resolution, $\pm 0.3\text{ }\mu\text{m}$ repeatability), and a KOHZU SA07A-RM02 electrode stage ($0.0012^\circ/0.0015^\circ$ about x/y). A measured 2.4-mm field of view gives an image-plane scale of $2.4/2048 = 0.001171875\text{ mm/px}$, used only for local image residuals.

Success requires $\text{TRE} < 0.50\text{ mm}$ for spatial registration and positional error $< 0.10\text{ mm}$ for the controller-internal target-localization stage. `ALIGN_NEEDLE` requires orientation error $< 2^\circ$ and image-plane error $< 0.005\text{ mm}$; `INSERT` requires orientation error $< 5^\circ$ and three-dimensional landing error $< 0.030\text{ mm}$. These thresholds evaluate the corresponding controller-level subtasks and are not used to define full-task completion. Spatial registration, target localization, image adjustment, and safety stopping use phantoms, calibration targets, and the robot platform without animal material. Only the physical `INSERT` subtask and complete closed-loop implantation trials used legally sourced ex-vivo

rat brain tissue obtained under institutional ethical approval and reported in accordance with the ARRIVE guidelines [22].

V. RESULTS AND ANALYSIS

A. Offline Model Comparison

All models are compared under one offline protocol. General VLMs without SFT estimate the zero-shot ceiling of generic multimodal ability; general VLMs with SFT quantify the benefit of state-skill labels; surgical VLMs test whether domain pretraining can replace task adaptation; and SurgiCLAW uses SFT followed by GDPO. Every model receives identical images, states, and queries and must use the same structured interface, which imposes a strict output format; prompt-only results therefore partly reflect adaptation to this interface rather than task ability alone.

On the fixed augmented test set, SurgiCLAW reaches 94.6% accuracy and 93.0% Macro-F1, improving over the strongest SFT general model by 5.5/6.5 points and over Qwen3-VL-4B-Instruct + SFT by 11.6/11.9 points, while its safety-violation rate drops to 0.5% (13/2,529) versus 3.2% and 1.9% for the 4B and 32B SFT models. Prompt-only general VLMs reach only 14.2–38.4% accuracy (Qwen3-VL-235B strongest at 31.2% Macro-F1), indicating that scale alone does not solve the state-to-skill mapping. After SFT the 4B model rises to 83.0%/81.1%, the 32B to 89.1%/86.5%, and surgical-specific VLMs remain below SFT, showing that domain semantics do not replace interface adaptation. Because augmented samples derived from the same original ID are correlated, these offline results are descriptive point estimates without confidence intervals or significance tests.

B. Real-Robot Comparison

Real-robot experiments evaluate the complete loop of high-level selection, deterministic control, and state update. SurgVLM-7B, SurgVidLM, Qwen3-VL-4B-Instruct + SFT, Qwen3-VL-32B-Instruct + SFT, and SurgiCLAW receive the same microscopic images, robot states, and task queries and share the same parser, action vocabulary, and low-level controllers. Each method is evaluated on 50 paired initial conditions, with randomized method order to reduce platform drift. Fig. 3 illustrates a representative closed-loop decision sequence, showing how SurgiCLAW maps successive image and robot-state observations to atomic-skill calls.

Table III reports closed-loop results. The safety-violation denominator is the number of safety-critical decision opportunities across trials, which differs by method because failed or repeated loops expose different numbers of such states; every violation is blocked by the parser. SurgiCLAW completes 44/50 trials (Wilson 95% interval 76.2–94.4%), improving over Qwen3-VL-4B-Instruct + SFT by 22 points (33/50 to 44/50) with 19.8% fewer stages and takeover falling from 20.0% to 4.0%; a paired McNemar test over this pair is statistically significant ($p < 0.05$). Relative to Qwen3-VL-32B-Instruct + SFT, it improves by 12 points (38/50 to 44/50), but this difference is not statistically significant (two-sided

McNemar $p = 0.070$), so we treat the gain over the 32B model as a trend rather than a confirmed improvement at 50 trials. Because all methods share the same low-level controllers, differences arise from the high-level policy triggering registration, alignment, insertion, recovery, and stopping at the right state: the surgical VLMs delay registration, repeat alignment, or request insertion before prerequisites hold, while the 32B SFT model is more robust than the 4B model, which is more sensitive to ambiguous visual states and recovery decisions. Failure categories are high-level selection, repeated alignment without convergence, visual anomaly/occlusion, safety-parser refusal, and insertion-control failure; total failures are 27/25/17/12/6, and SurgiCLAW’s six failures break down as 1/2/1/1/1, so repeated alignment under occlusion and physical insertion remain the principal bottlenecks.

C. Ablation Study

Ablations remove only model-side training or output components, while keeping low-level controllers, data splits, and metrics unchanged. All variants are evaluated on the same fixed test set with accuracy, macro-precision, macro-recall, Macro-F1, and safety violation.

Table IV shows the ablation results. Removing the structured rationale lowers accuracy/Macro-F1 by 6.4/6.6 points, and the SFT-only variant (w/o GDPO, equal to Qwen3-VL-4B-Instruct + SFT) remains far above prompt-only baselines but below the full model, indicating that supervised skill labels alone do not fully cover boundary and safety-critical states: adding GDPO raises accuracy from 83.0% to 94.6%, Macro-F1 from 81.1% to 93.0%, and cuts safety violation from 3.2% to 0.5%. Removing the action-correctness reward mainly degrades class-level selection, removing the prerequisite reward gives the highest safety-violation rate (confirming its role in constraining state/function conflicts), and removing the format reward mainly lowers recall. GDPO therefore improves accuracy, parseability, and safety-aware selection.

D. Controller-Level Subtask Verification

Each deterministic controller was tested 40 times under fixed perturbations, independently of the large-model decisions, following the frameworks of [14]–[16].

Table V reports controller-level success and accuracy. All controllers succeed on 90.0%–100.0% of trials; the spatial-registration TRE of 0.347 ± 0.126 mm, needle-alignment error of $0.91 \pm 0.42^\circ$, and insertion error of 0.015 ± 0.006 mm fall within the corresponding thresholds. Errors and times are means and standard deviations over successful trials.

VI. CONCLUSION

This paper presented SurgiCLAW, a hierarchical atomic-skill interface for precision robotic micromanipulation. Its central idea is to keep the semantic decision making of a multimodal language model within a small, bounded, and auditable skill space, and to gate every decision through a safety parser before delegating execution to verified deterministic controllers; this separates *what* to do next from *how precisely* to do it. It also makes the interface learnable from limited

TABLE II
ATOMIC-SKILL SELECTION ON THE FIXED TEST SET

Method	Accuracy (%)↑	Macro-P (%)↑	Macro-R (%)↑	Macro-F1 (%)↑	Safety violation (%)↓
<i>General VLMs (non-SFT)</i>					
MiniCPM-o-2.6 [23]	14.2	2.0	14.3	3.6	18.6
DeepSeek-VL2 [24]	20.6	22.4	20.9	14.8	14.2
Llama-4-Maverick-17B-128E-Instruct [25]	24.7	28.9	25.1	17.2	11.6
InternVL3.5-241B-A28B [26]	31.6	37.8	31.2	24.4	7.4
Qwen2.5-VL-3B-Instruct [27]	15.4	27.8	15.5	5.9	15.8
Qwen2.5-VL-7B-Instruct [27]	28.0	29.7	28.2	18.2	10.9
Qwen3-VL-4B-Instruct [17]	30.0	40.8	30.3	23.8	7.8
Qwen3-VL-30B-A3B-Instruct [17]	35.0	44.1	35.2	27.6	5.9
Qwen3-VL-235B-A22B-Thinking [17]	38.4	43.6	38.0	31.2	4.8
<i>Surgical-specific VLMs</i>					
SurgVLM-7B [12]	41.8	47.2	39.7	38.6	5.3
SurgVidLM [13]	44.6	49.1	43.8	42.7	4.9
<i>General VLMs + SFT</i>					
Qwen3-VL-4B-Instruct + SFT	83.0	81.4	81.0	81.1	3.2
Qwen3-VL-32B-Instruct + SFT	89.1	87.6	85.6	86.5	1.9
DeepSeek-VL2-Small + SFT	81.7	78.7	78.0	78.0	4.2
<i>SFT + GDPO</i>					
SurgiCLAW	94.6	94.0	92.0	93.0	0.5

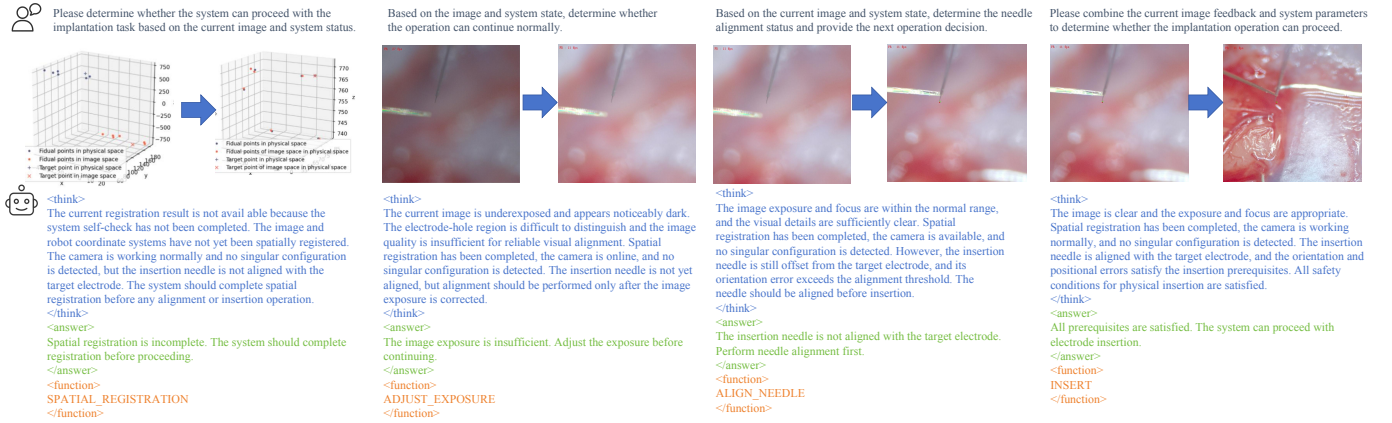


Fig. 3. Closed-loop decision example for flexible-electrode implantation. At each step, SurgiCLAW receives the current microscopic image, robot state, and operation query, then outputs a structured rationale, an operator-facing answer, and a single atomic-skill call. The example illustrates a typical decision sequence from spatial registration and image-quality correction to needle alignment and final insertion.

surgical data: a multi-MLLM data-generation pipeline expands 5,000 state-skill samples into 30,000 structured ones, and a three-reward group reward-decoupled normalization policy optimization objective improves skill selection, prerequisite safety, and output parseability. On the fixed test set SurgiCLAW reaches 94.6% accuracy and 93.0% Macro-F1, and it completes 88.0% of 50 closed-loop implantation trials. We interpret these results as a feasibility demonstration rather than evidence of clinical efficacy: the evaluation covers a single high-precision task using phantom and ex-vivo material, and the closed-loop comparisons are constrained by the modest number of paired trials. Whether the safety and interpretability benefits generalize to other tasks, to more stringent clinical conditions, or to long-horizon unattended operation remains to be established.

REFERENCES

- [1] K. Kawaharazuka, J. Oh, J. Yamada, I. Posner, and Y. Zhu, "Vision-language-action models for robotics: A review towards real-world applications," *IEEE Access*, 2025.
- [2] P. Intelligence, K. Black, N. Brown, J. Darpinian, K. Dhabalia, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai *et al.*, "pi05: a vision-language-action model with open-world generalization," *arXiv preprint arXiv:2504.16054*, 2025.
- [3] M. Xu, Z. Huang, J. Zhang, X. Zhang, and Q. Dou, "Surgical action planning with large language models," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2025, pp. 563–572.
- [4] S.-Y. Liu, X. Dong, X. Lu, S. Diao, P. Belcak, M. Liu, M.-H. Chen, H. Yin, Y.-C. F. Wang, K.-T. Cheng *et al.*, "Gdpo: Group reward-decoupled normalization policy optimization for multi-reward rl optimization," *arXiv preprint arXiv:2601.05242*, 2026.
- [5] M. Ahn, A. Brohan, N. Brown, Y. Chebotar, O. Cortes, B. David, C. Finn, C. Fu, K. Gopalakrishnan, K. Hausman *et al.*, "Do as i can, not as i say: Grounding language in robotic affordances," *arXiv preprint arXiv:2204.01691*, 2022.
- [6] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn *et al.*, "Rt-2: Vision-language-action models transfer web knowledge to robotic control," *arXiv preprint arXiv:2307.15181*, 2023.
- [7] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi *et al.*, "Openvla: An open-source vision-language-action model," *arXiv preprint arXiv:2406.09246*, 2024.
- [8] A. Shademan, R. S. Decker, J. D. Opfermann, S. Leonard, A. Krieger,

TABLE III
CLOSED-LOOP REAL-ROBOT IMPLANTATION RESULTS

Method	Success (%)↑	Stages↓	Takeover (%)↓	Unsafe proposal (%)↓
<i>Surgical-specific VLMs</i>				
SurgVLM-7B	46.0 (23/50)	10.4 ± 2.6	38.0 (19/50)	5.4 (11/204)
SurgVidLM	50.0 (25/50)	9.8 ± 2.4	34.0 (17/50)	4.9 (10/205)
<i>General VLM + SFT</i>				
Qwen3-VL-4B-Instruct + SFT	66.0 (33/50)	8.6 ± 1.9	20.0 (10/50)	3.4 (7/207)
Qwen3-VL-32B-Instruct + SFT	76.0 (38/50)	7.7 ± 1.6	10.0 (5/50)	1.9 (4/206)
<i>SFT + GDPO</i>				
SurgiCLAW	88.0 (44/50)	6.9 ± 1.5	4.0 (2/50)	0.5 (1/194)

TABLE IV
ABLATION OF MODEL-SIDE COMPONENTS

Setting	Accuracy (%)↑	Macro-P (%)↑	Macro-R (%)↑	Macro-F1 (%)↑	Safety violation (%)↓
Full model	94.6	94.0	92.0	93.0	0.5
w/o structured rationale	88.2	87.4	85.9	86.4	2.3
w/o GDPO	83.0	81.4	81.0	81.1	3.2
w/o action-correctness reward	90.4	88.1	88.7	88.2	1.6
w/o prerequisite reward	91.2	90.8	88.9	89.7	3.2
w/o format reward	92.0	92.8	88.6	90.4	1.1

TABLE V
ATOMIC-SUBTASK SUCCESS AND EXECUTION ACCURACY

Subtask	Success criterion	Success (%)↑	Mean error↓	Time (s)↓
Spatial registration	TRE < 0.50 mm and coordinate transform completed	95.0 (38/40)	0.347 ± 0.126 mm (TRE)	3.18 ± 0.37
Target localization (internal)	Stable target detection available for alignment	97.5 (39/40)	0.051 ± 0.028 mm (position)	0.82 ± 0.26
Needle alignment	Orientation and image-plane position within thresholds	95.0 (38/40)	0.91 ± 0.42° / 0.0018 ± 0.0010 mm	19.4 ± 4.8
Insertion execution	Direction and 3-D landing position within thresholds	90.0 (36/40)	2.57 ± 1.16° / 0.015 ± 0.006 mm	45.73 ± 13.49
Image-quality adjustment	Image restored to a parseable quality range	97.5 (39/40)	–	1.63 ± 0.61
Safety stop	Timely refusal in a high-risk state	100.0 (40/40)	–	0.12 ± 0.02

- and P. C. Kim, “Supervised autonomous robotic soft tissue surgery,” *Science translational medicine*, vol. 8, no. 337, pp. 337ra64–337ra64, 2016.
- [9] J. W. Kim, T. Z. Zhao, S. Schmidgall, A. Deguet, M. Kobilarov, C. Finn, and A. Krieger, “Surgical robot transformer (srt): Imitation learning for surgical tasks,” *arXiv preprint arXiv:2407.12998*, 2024.
- [10] J. W. Kim, J.-T. Chen, P. Hansen, L. X. Shi, A. Goldenberg, S. Schmidgall, P. M. Scheikl, A. Deguet, B. M. White, D. R. Tsai *et al.*, “Srt-h: A hierarchical framework for autonomous surgery via language-conditioned imitation learning,” *Science robotics*, vol. 10, no. 104, p. eadt5254, 2025.
- [11] M. Moghani, L. Doorenbos, W. C.-H. Panitch, S. Huver, M. Azizian, K. Goldberg, and A. Garg, “Sufia: language-guided augmented dexterity for robotic surgical assistants,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 6969–6976.
- [12] Z. Zeng, Z. Zhuo, X. Jia, E. Zhang, J. Wu, J. Zhang, Y. Wang, C. H. Low, J. Jiang, Z. Zheng *et al.*, “Surgvlm: A large vision-language model and systematic evaluation benchmark for surgical intelligence,” *arXiv preprint arXiv:2506.02555*, 2025.
- [13] G. Wang, J. Wang, W. Mo, L. Bai, K. Yuan, M. Hu, J. Wu, J. He, Y. Huang, N. Padoy *et al.*, “Surgvidlm: Towards multi-grained surgical video understanding with large language model,” *arXiv preprint arXiv:2506.17873*, 2025.
- [14] A. Ye, Y. Song, J. Su, and D. Zhang, “Non-invasive spatial registration using customized dental bracket and improved genetic algorithms,” in *2024 IEEE 20th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2024, pp. 3250–3255.
- [15] A. Ye, X. Wang, Y. Song, Y. Li, Z. Zhu, and D. Zhang, “Fully automated insertion algorithm of flexible electrodes for invasive brain-machine interface,” *Pattern Recognition Letters*, 2026.
- [16] A. Ye, X. Wang, Y. Song, Y. Li, S. Yu, and D. Zhang, “Holenet: Neural network for 5-dof skull-hole-center localization for minimally invasive brain-machine interfaces surgery,” *IEEE Transactions on Medical Robotics and Bionics*, 2026.
- [17] S. Bai, Y. Cai, R. Chen, K. Chen, X. Chen, Z. Cheng, L. Deng, W. Ding, C. Gao, C. Ge *et al.*, “Qwen3-vl technical report,” *arXiv preprint arXiv:2511.21631*, 2025.
- [18] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen *et al.*, “Lora: Low-rank adaptation of large language models,” *Iclr*, vol. 1, no. 2, p. 3, 2022.
- [19] Y. Zhao, A. Gu, R. Varma *et al.*, “PyTorch FSDP: Experiences on scaling fully sharded data parallel,” *Proceedings of the VLDB Endowment*, vol. 16, no. 12, pp. 3848–3860, 2023.
- [20] E. B. Wilson, “Probable inference, the law of succession, and statistical inference,” *Journal of the American Statistical Association*, vol. 22, no. 158, pp. 209–212, 1927.
- [21] Q. McNemar, “Note on the sampling error of the difference between correlated proportions or percentages,” *Psychometrika*, vol. 12, no. 2, pp. 153–157, 1947.
- [22] N. Percie du Sert, V. Hurst, A. Ahluwalia *et al.*, “The ARRIVE guidelines 2.0: Updated guidelines for reporting animal research,” *PLOS Biology*, vol. 18, no. 7, p. e3000410, 2020.
- [23] Y. Yao, T. Yu, A. Zhang, C. Wang, J. Cui, H. Zhu, T. Cai, H. Li, W. Zhao, Z. He *et al.*, “Minicpm-v: A gpt-4v level mllm on your phone,” *arXiv preprint arXiv:2408.01800*, 2024.
- [24] Z. Wu, X. Chen, Z. Pan, X. Liu, W. Liu, D. Dai, H. Gao, Y. Ma, C. Wu, B. Wang *et al.*, “Deepseek-vl2: Mixture-of-experts vision-language models for advanced multimodal understanding,” *arXiv preprint arXiv:2412.10302*, 2024.
- [25] Meta AI, “The Llama 4 herd: The beginning of a new era of natively multimodal AI innovation,” Online, 2025, accessed: 2026-08-07. [Online]. Available: <https://ai.meta.com/blog/llama-4-multimodal-intelligence/>
- [26] W. Wang *et al.*, “InternVL3.5: Advancing open-source multimodal

models in versatility, reasoning, and efficiency,” *arXiv preprint arXiv:2508.18265*, 2025.

- [27] Qwen Team, “Qwen2.5-VL technical report,” *arXiv preprint arXiv:2502.13923*, 2025.