Data-Driven Detection of Executional Errors in Robot-Assisted Surgery

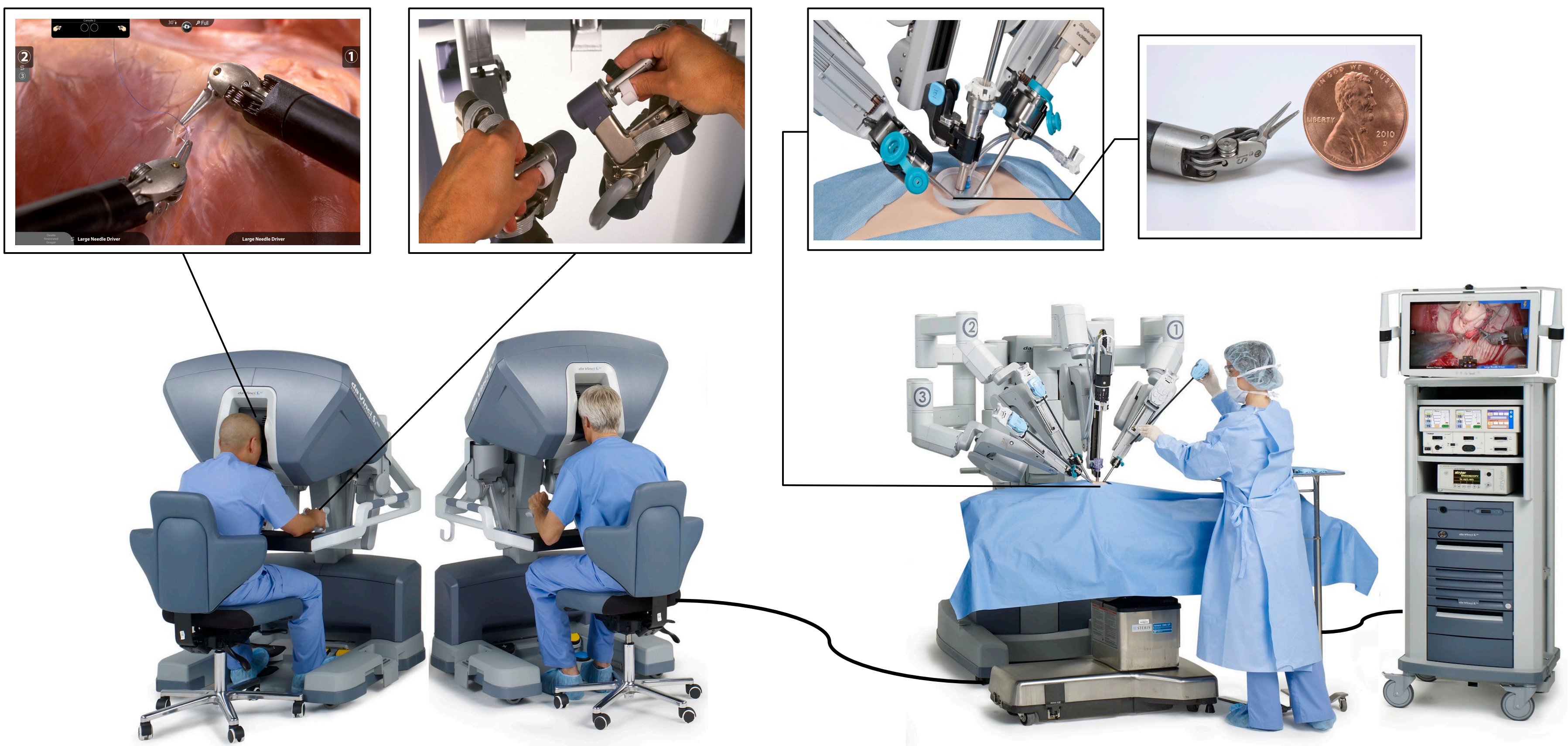
Homa Alemzadeh
Dependable Systems and Analytics Group
Electrical and Computer Engineering
CPS Link Lab

UVA LINK LAB

UNIVERSITY of VIRGINIA

UVA DSA

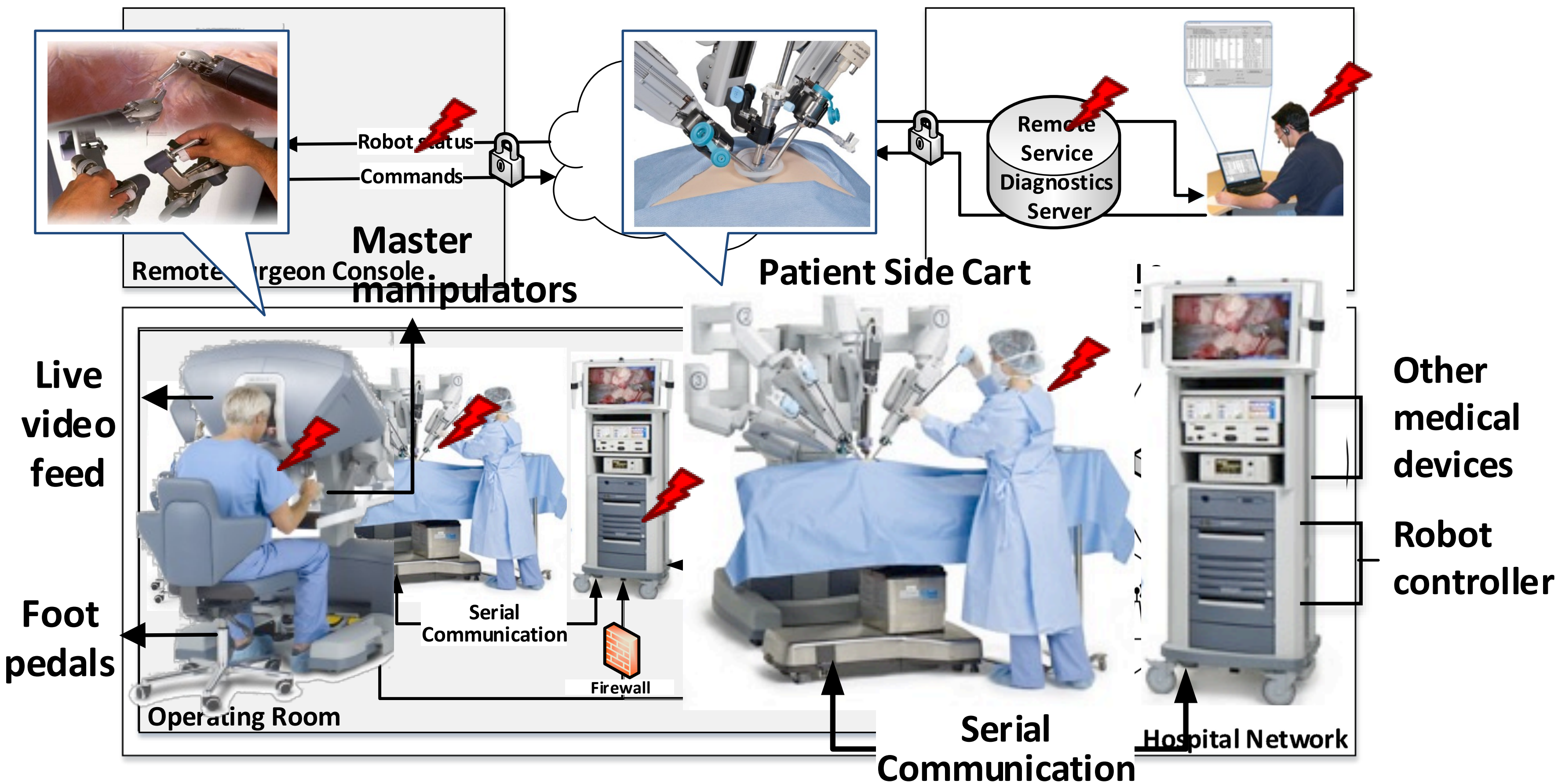
Robot-Assisted Minimally Invasive Surgery



Loosely Closed-loop Semi-autonomous:
No haptics, limited vision feedback

Credits: da Vinci Surgical System, Intuitive Surgical, Inc.

Human-Cyber-Physical Systems

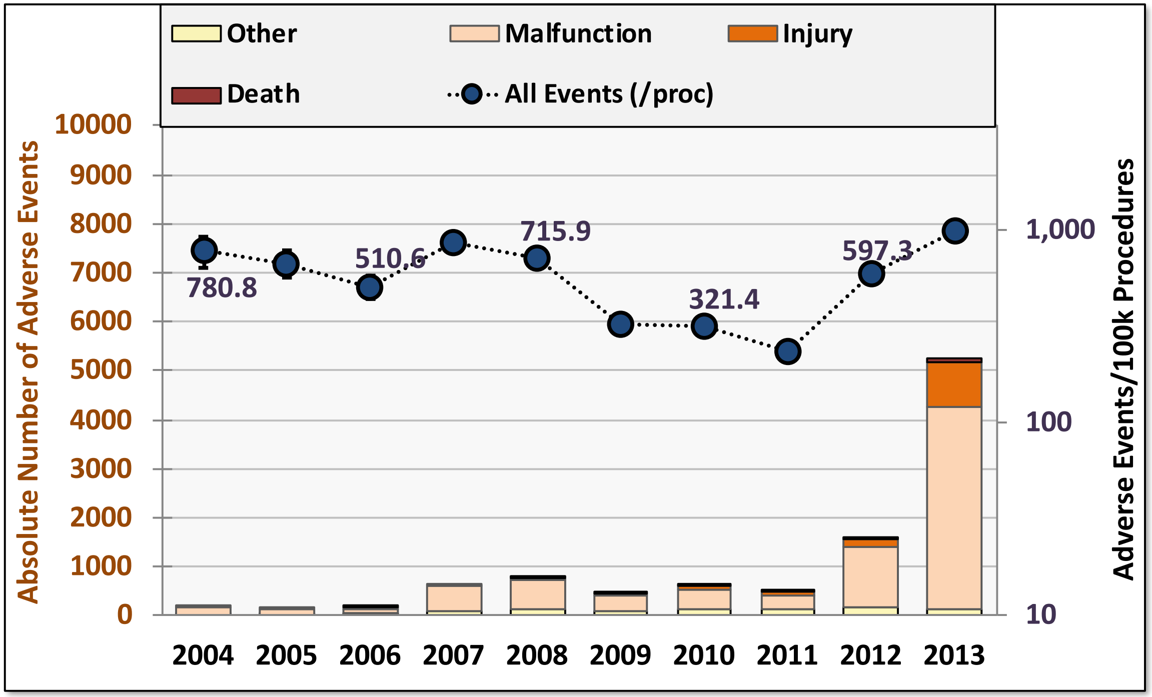


Adverse Events in Robotic Surgery

Once in every 100 robotic procedures an unexpected adverse event is likely to happen.

Example Adverse Events:

- Unexpected cuts, bleeding, minor injuries, other complications
- Long procedure times spent on troubleshooting system errors
- Converting/rescheduling procedure

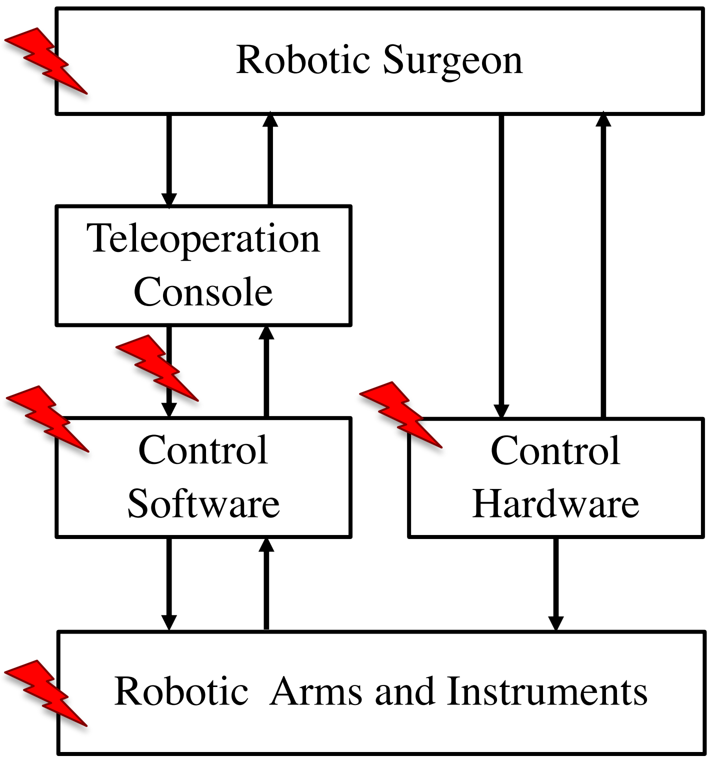


Alemzadeh, et al. "Adverse Events in Robotic Surgery: A Retrospective Study of 14 Years of FDA Data". PLOS ONE, 2016.

Adverse Events in Robotic Surgery

Erroneous or unexpected robotic movements:

- Incorrect robot end-effector positions and orientations
- Incorrect grasper angle or force

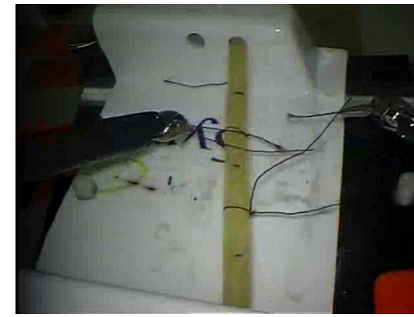


Unintended Human Errors

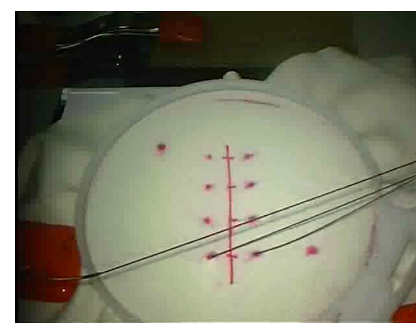
DOS and MITM Attacks *[ICCPS 2015]*

Faulty firewall *[WIRED 2014]*

Malware targeting control software *[DSN 2016]*



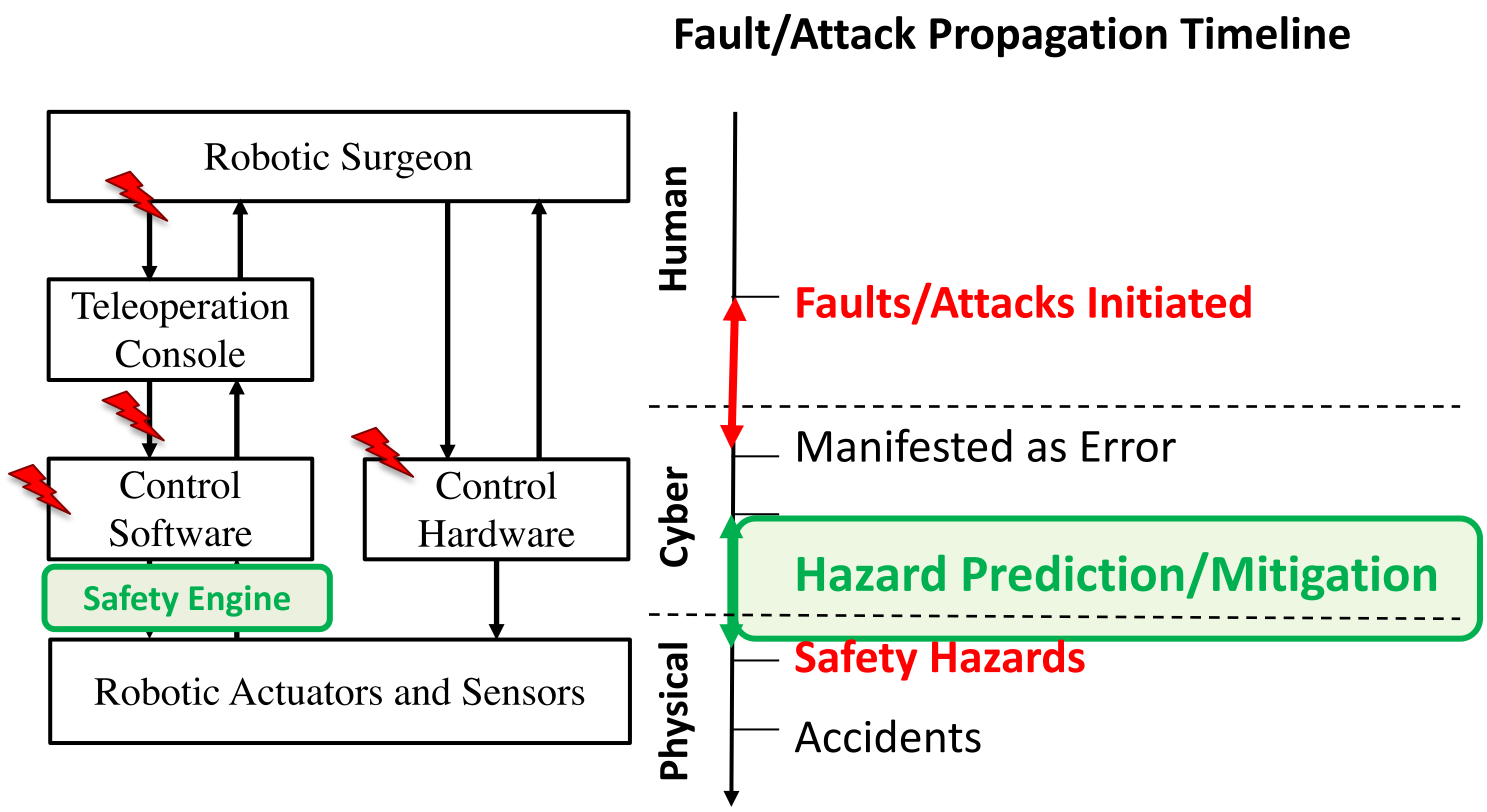
High Force



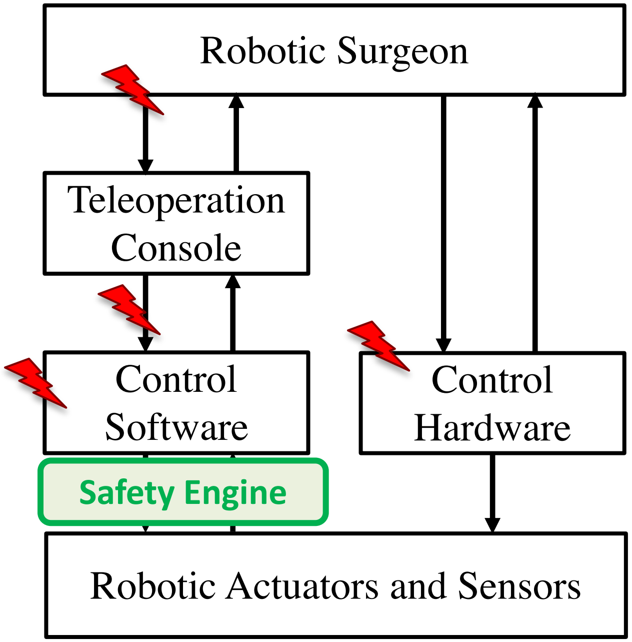
Graspers out of sight

Credits: Gao, Yixin, et al., MICCAI Workshop: M2CAI. Vol. 3. 2014.

Safety Engine for Runtime Monitoring



Safety Engine for Runtime Monitoring



- **Preemptive Detection of Safety Hazards**
  Unsafe system context leading to unsafe control actions

- **Early context-specific feedback to surgeons**
  Potentially sub-optimal or unsafe motions leading to low performance scores in training or safety-critical events in actual surgery

Surgical Skill Evaluation Techniques

- **Manual offline assessment**
  - OSATS, GOALS, GEARS, R-OSATS
  - Subjective, cognitively demanding, and prone to errors

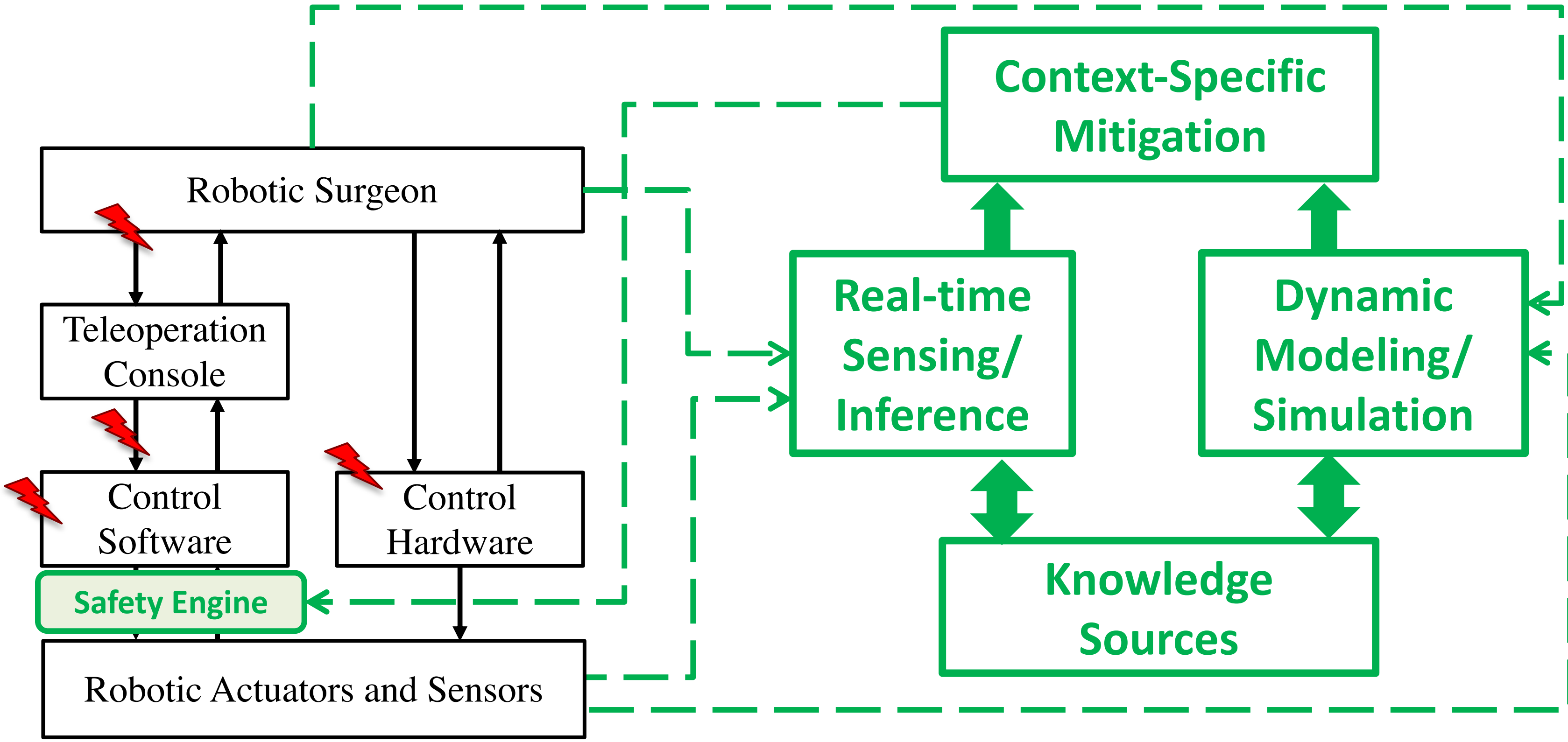
- **Automated online assessment**
  - Using kinematic, video, and system event data, focusing on subtasks
  - Efficiency metrics (e.g., path length, completion time)
  - Task/procedure specific metrics (e.g., camera movement, energy activation)
  - Safety metrics (e.g., instrument collisions, instruments out of view, excessive force, needle drops, tissue damage)
  - Less attention to task and gesture specific errors

Existing Anomaly Detection Solutions

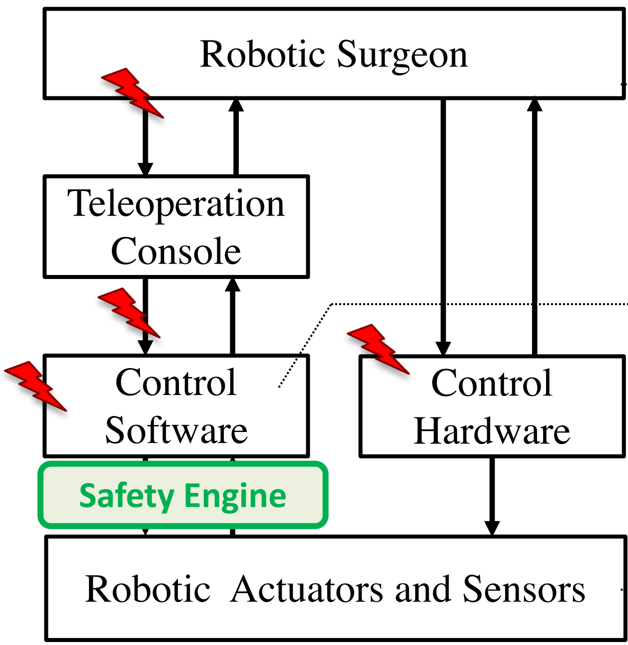
- **Offline safety and security assurance**
  - Risk analysis, model-driven design, synthesis, and formal verification
  - Residual faults, vulnerabilities, human errors still appear at runtime

- **Runtime monitoring and automated recovery**
  - Often rely on fixed rules and general medical guidelines
  - Combined cyber-physical monitoring requires modeling physical system, environment and complex patient dynamics
  - Medical workflows and the human operator’s actions overlooked
  - False alarms and late detection negatively affect correction and recovery
  - Solely data-driven or model-driven => small data, lack of transparency

Context-Aware Runtime Monitoring



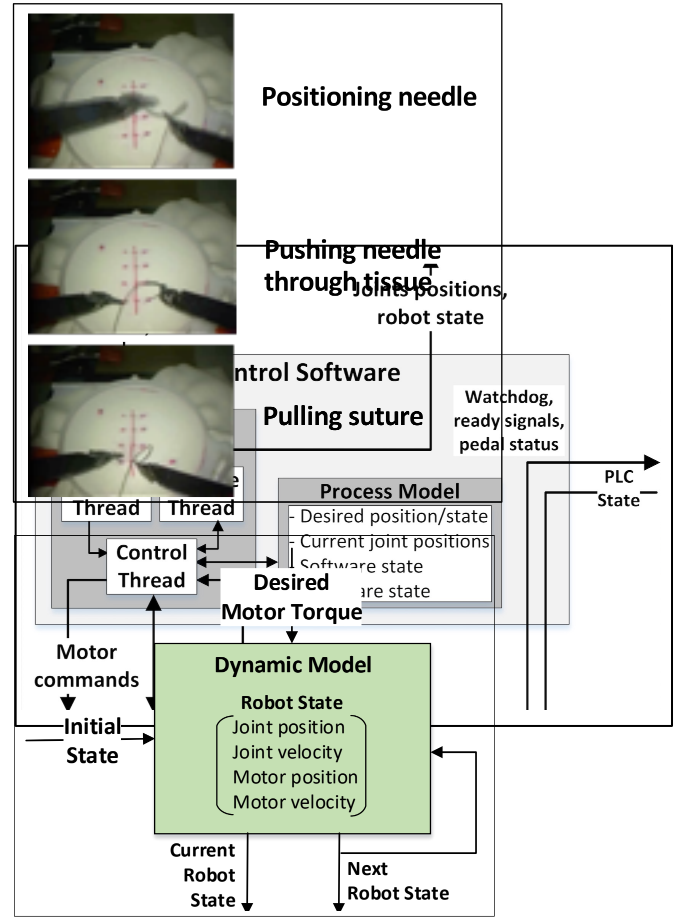
Safety Context in Surgery



Operational context
- Surgical Workflow
  - Surgical task, subtask

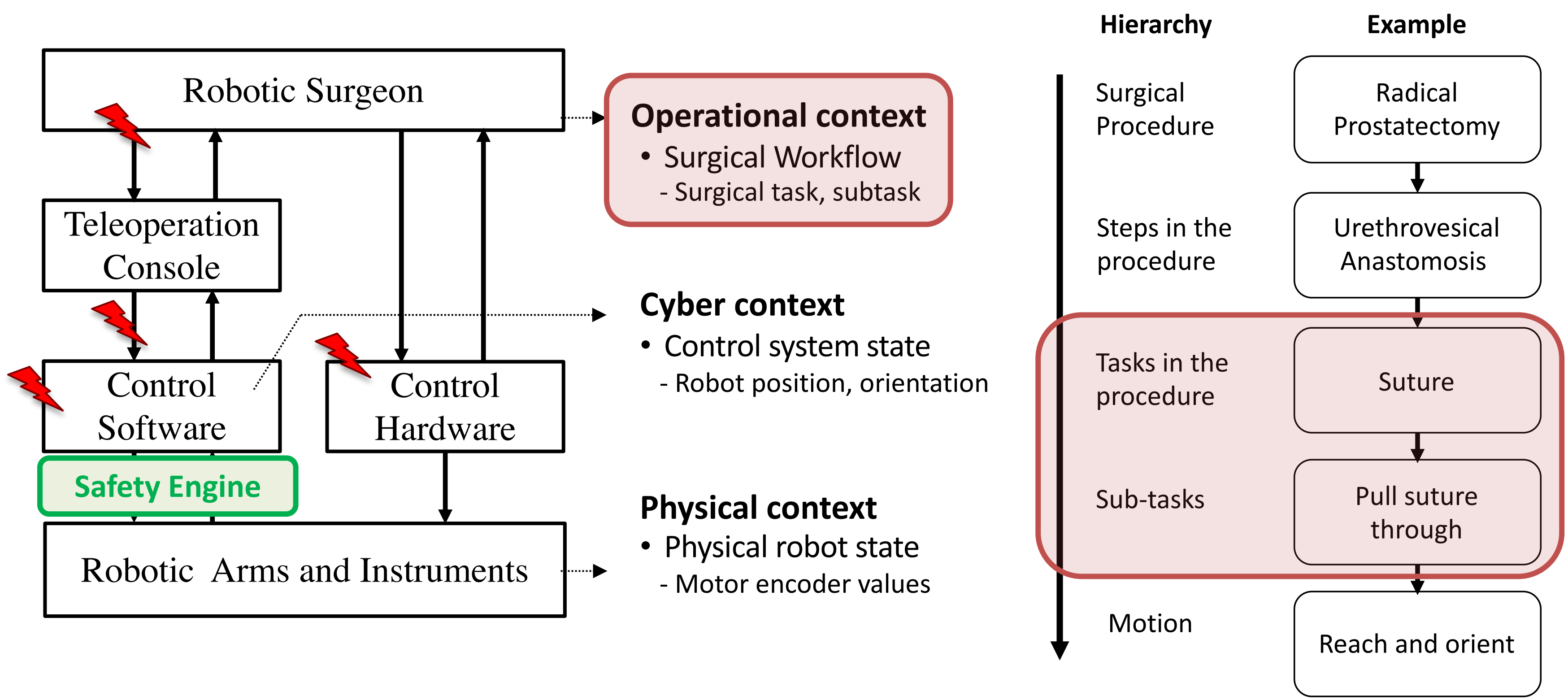
Cyber context
- Control system state
  - Robot position, orientation

Physical context
- Physical robot state
  - Motor encoder values

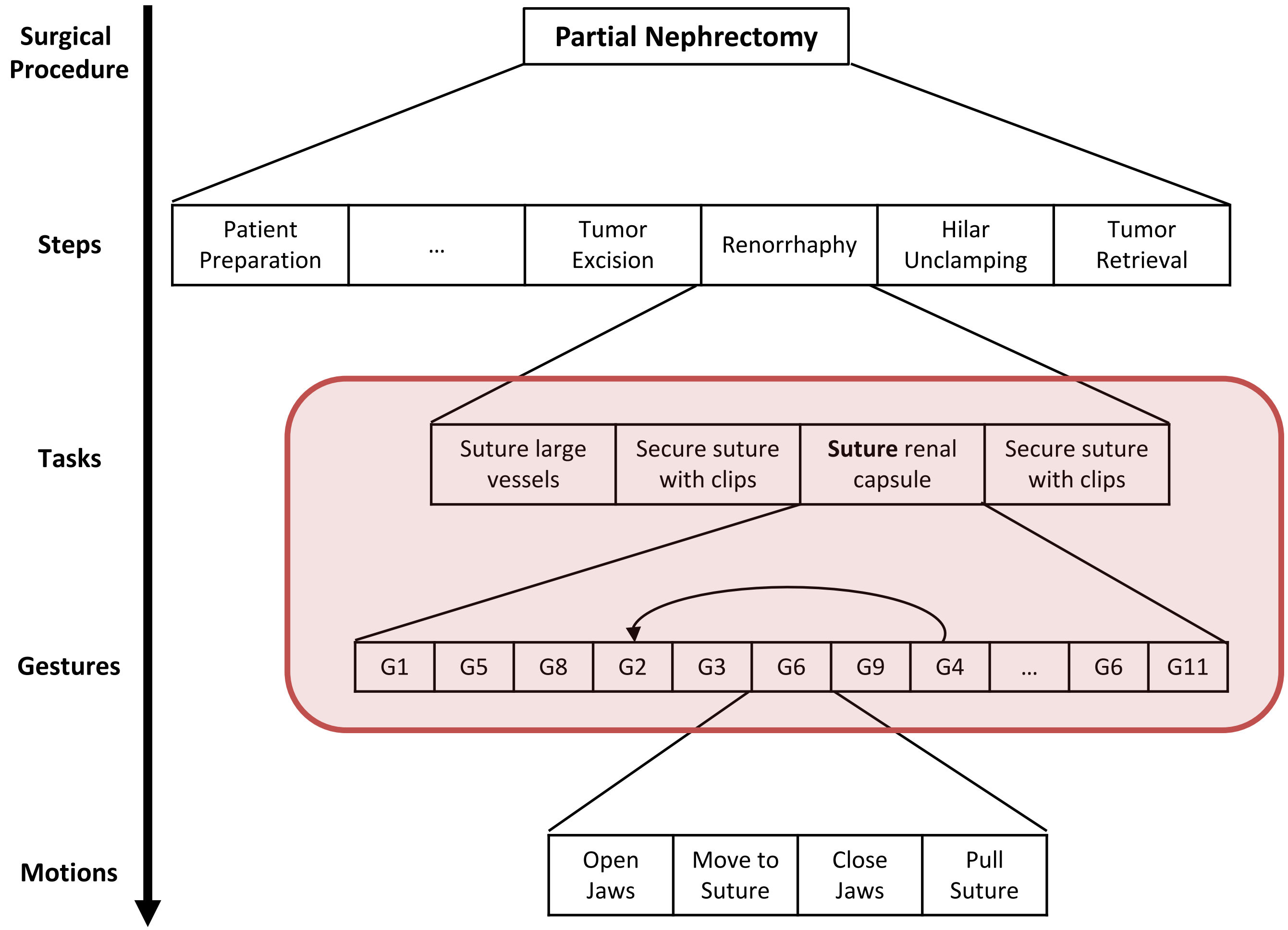


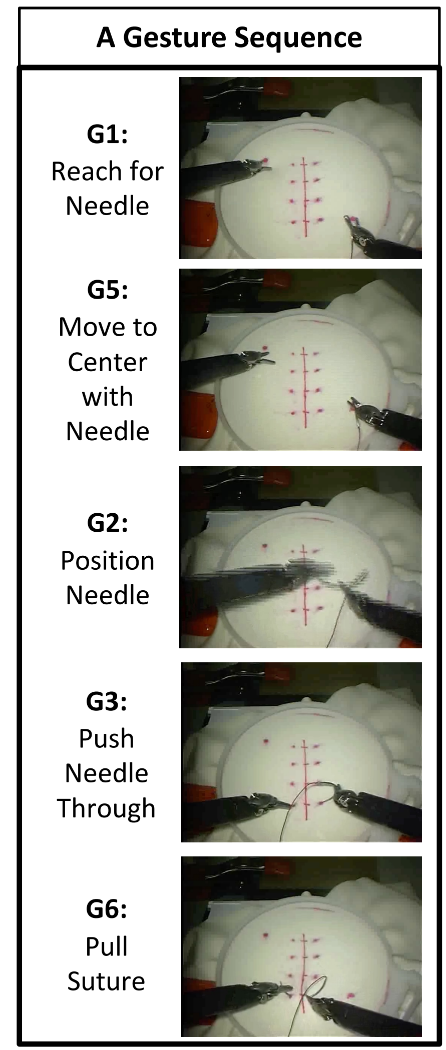
Alemzadeh, et al., "Targeted Attacks on Teleoperated Surgical Robots: Dynamic Model-Based Detection and Mitigation", DSN 2016.
Yasar at el., "Real-Time Context-aware Detection of Unsafe Events in Robot-Assisted Surgery", ISMR 2019, DSN 2020.

Operational Context in Surgery



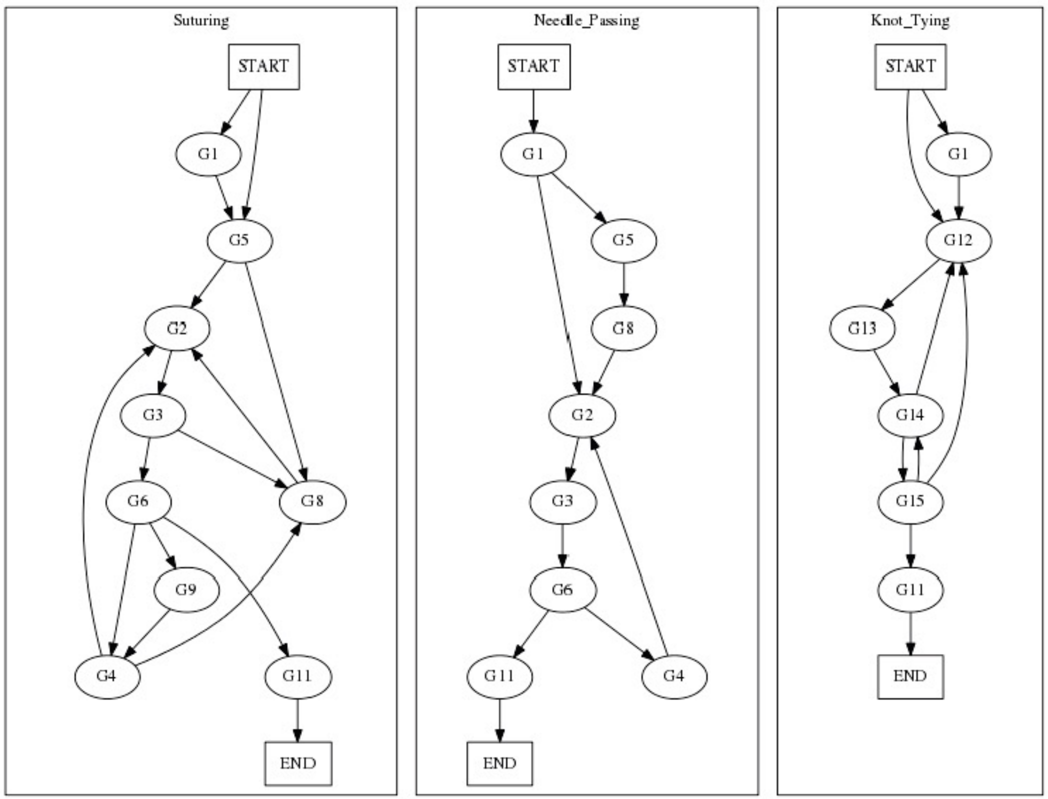
Operational Context in Surgery

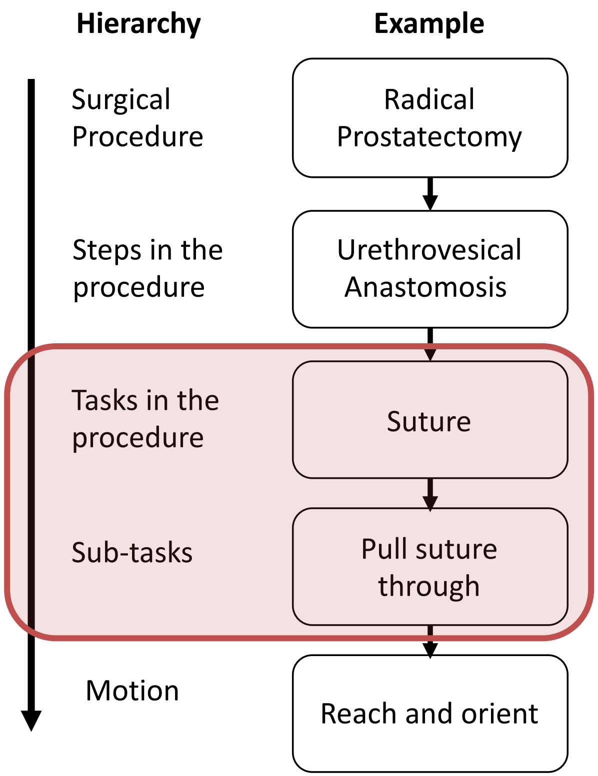




K. Hutchinson, et al.,"Analysis of Executional and Procedural Errors in Dry-lab Robotic Surgery Experiments," arXiv:2106.11962, 2021.

Task Grammar Graphs





N. Ahmidi et al, "A Dataset and Benchmarks for Segmentation and Recognition of Gestures in Robotic Surgery," IEEE Trans. Biomedical Eng., 2016.

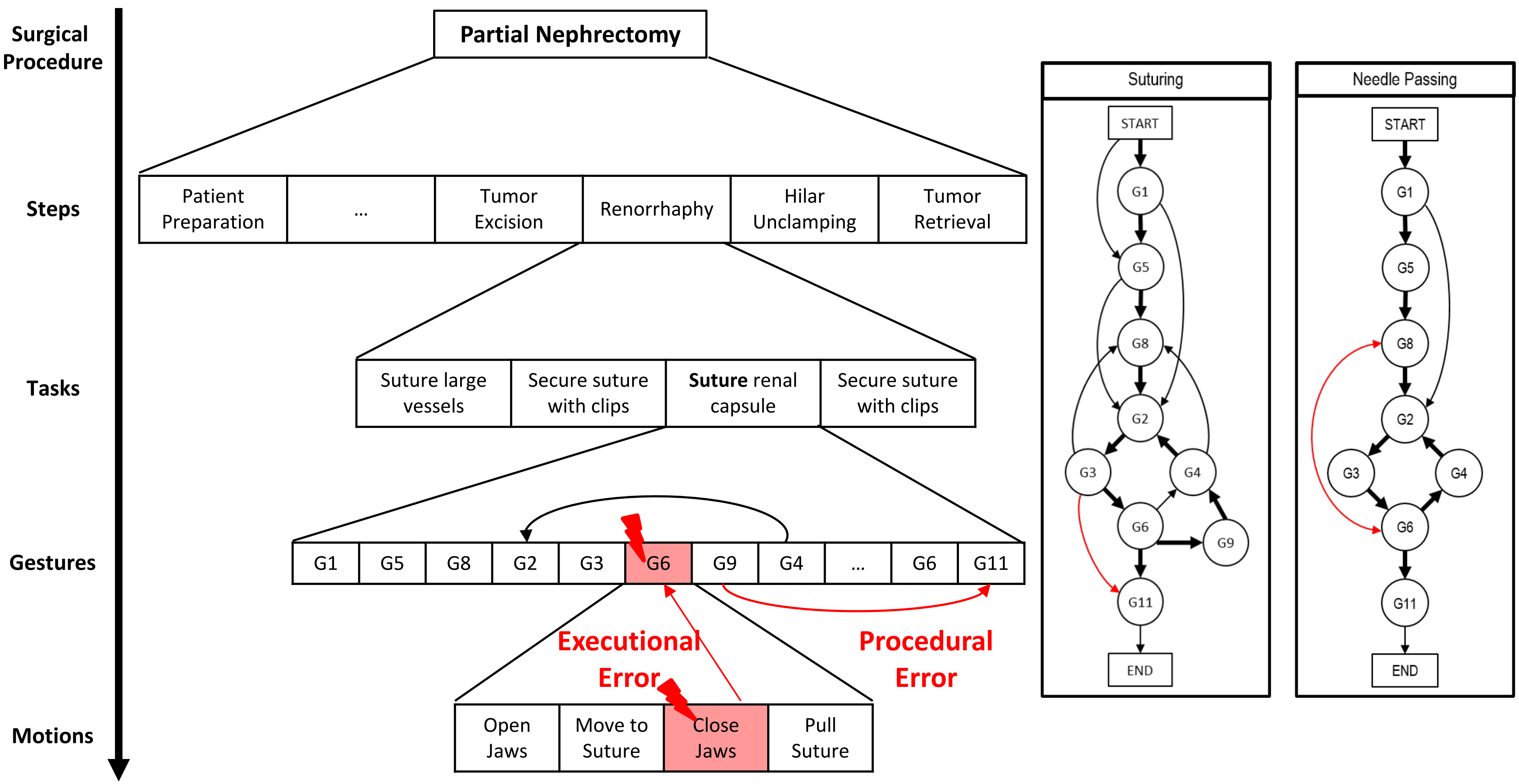
Error Rubrics for Surgery

- **Procedural errors:** the omission or re-arrangement of correctly undertaken steps within the procedure. (Categories 1–6)
- **Execution errors:** the failure in correctly executing an individual step.
  - Failure of a specific motor task within the procedure, e.g., performing an action with too much or too little force. (Categories 7–10)

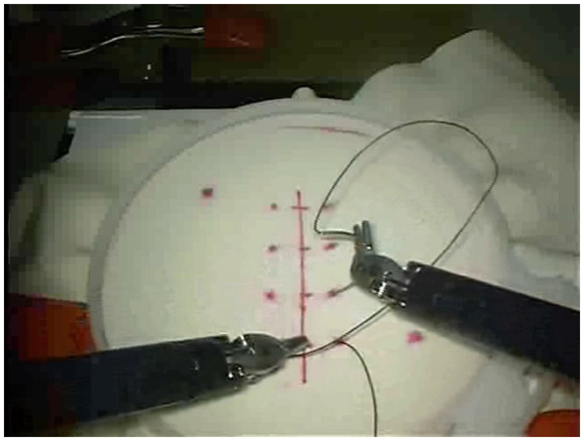
1. Step is *not done*
2. Step is *partially* completed
3. Step is *repeated*
4. Second step is done *in addition*
5. Second step is done *instead of* first step
6. Step is done *out of sequence*
7. Step is done with *too much* (speed, force, distance, time, rotation, depth)
8. Step is done with *too little* (speed, force, distance, time, rotation, depth)
9. Step is done in *wrong* (orientation, direction, point in space)
10. Step is done on/ with the *wrong object*

P. Joice, et al, "Errors enacted during endoscopic surgery--a human reliability analysis.," Appl. Ergon., 1998.

Context-Specific Executional & Procedural Errors

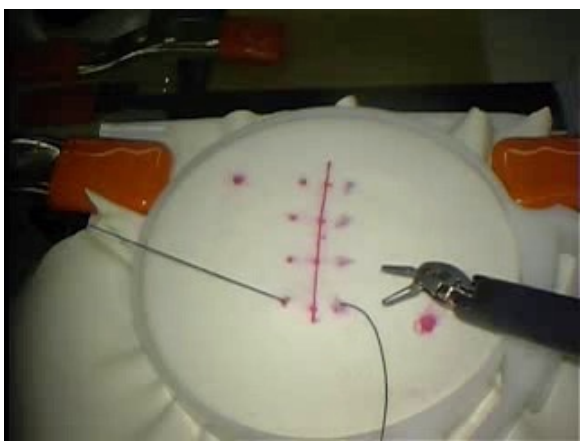


Analysis of Executional Errors in JIGSAWS

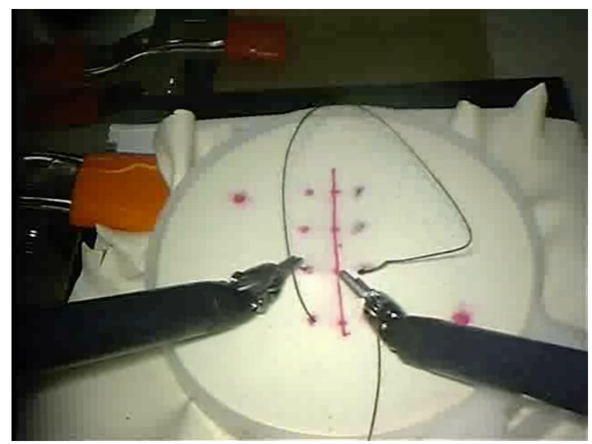


Multiple Attempts

Needle Drop



Needle Orientation

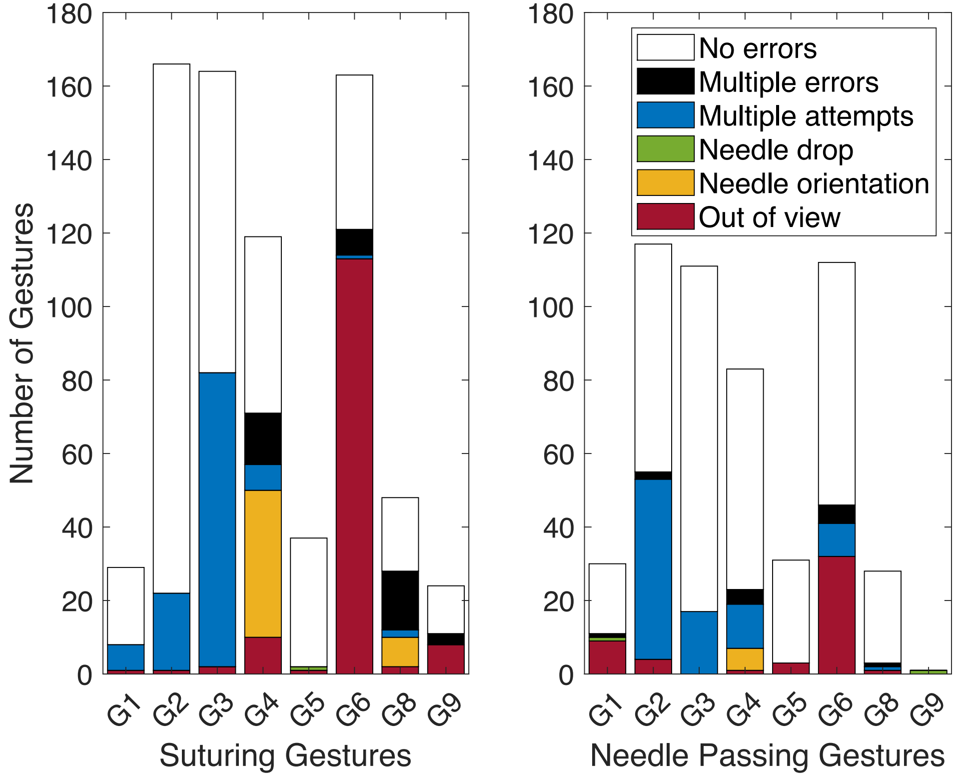


Out of View

https://github.com/UVA-DSA/ExecProc_Error_Analysis

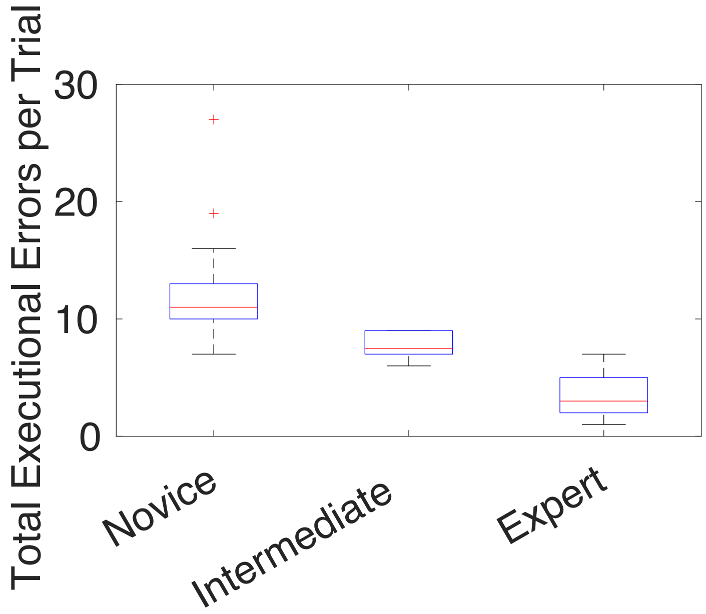
Gesture-Specific Executional Errors

- **More challenging gestures in each task are more prone to executional errors.**
  - G6, G3, and G4 in Suturing and G2 and G6 in Needle Passing.
- **Each gesture has a predominant executional error mode.**
  - G2 and G3: “Multiple attempts” errors, G6: “Out of view” errors

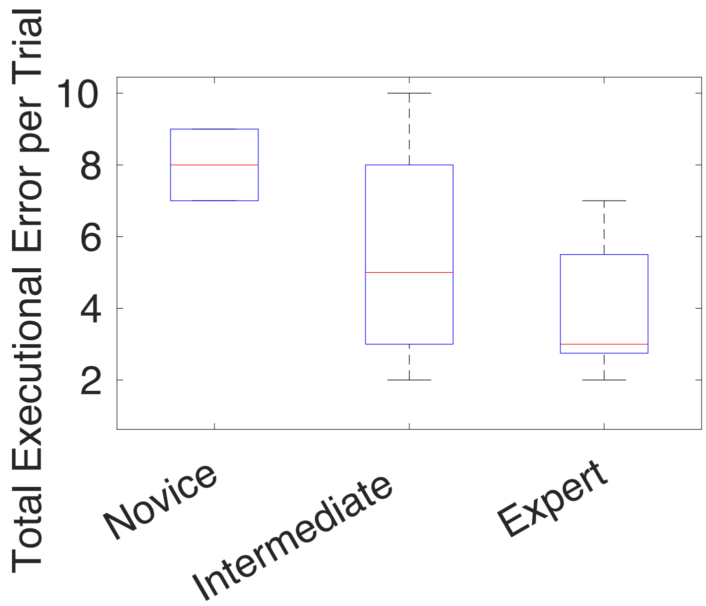


Executional Errors and Skill Levels

- **The total number of executional errors per trial often correlate with the skill levels**
  - Both self-proclaimed and GRS based.



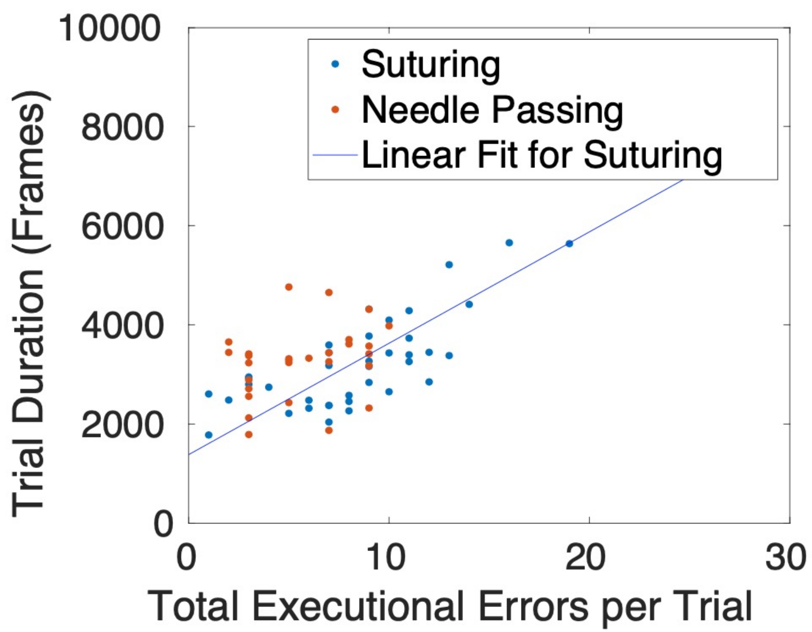
(a) Self-Proclaimed Skill Levels

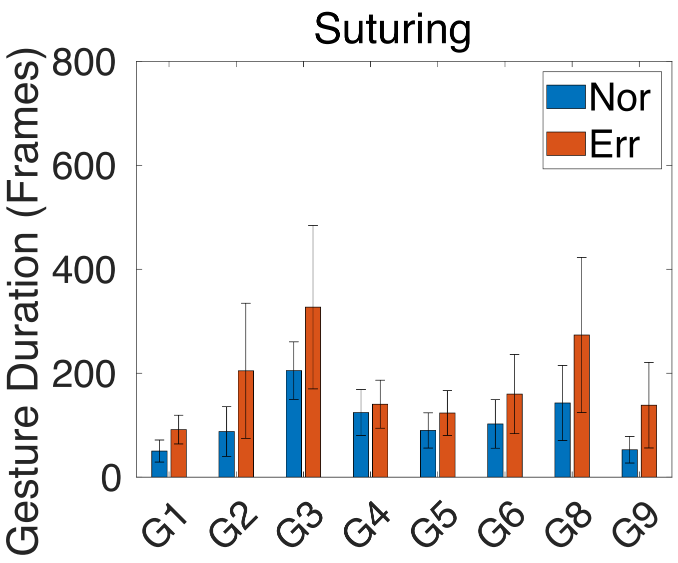


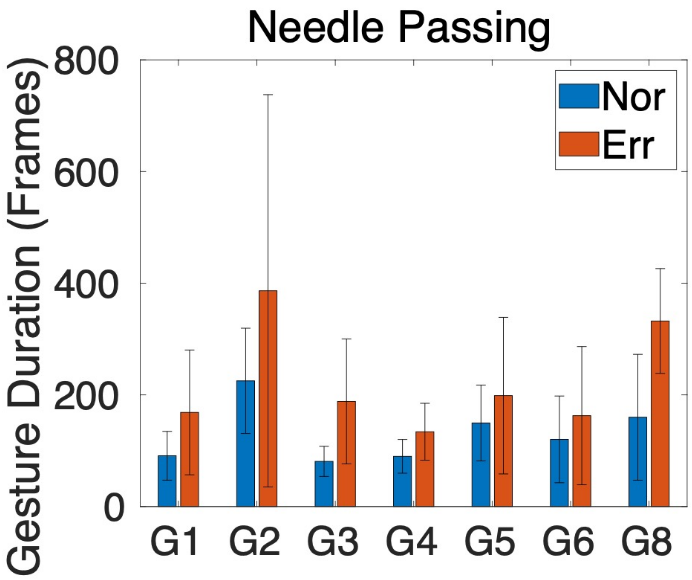
(b) GRS Skill Levels

Executional Errors and Trial Duration

- Both executional and procedural errors lead to lengthier trials, especially during more complicated tasks such as Suturing.



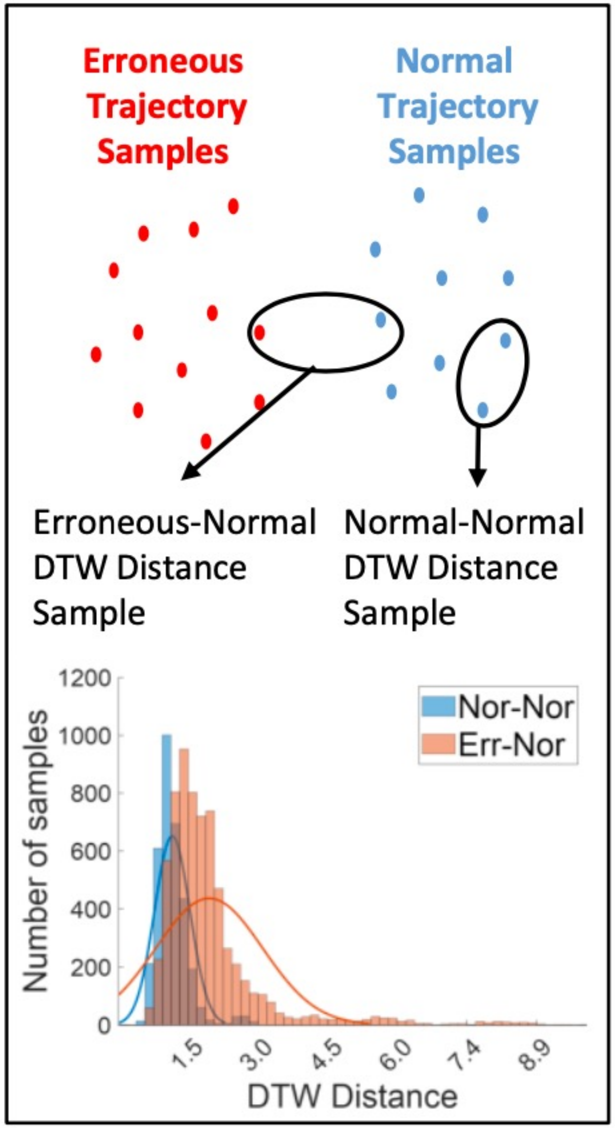




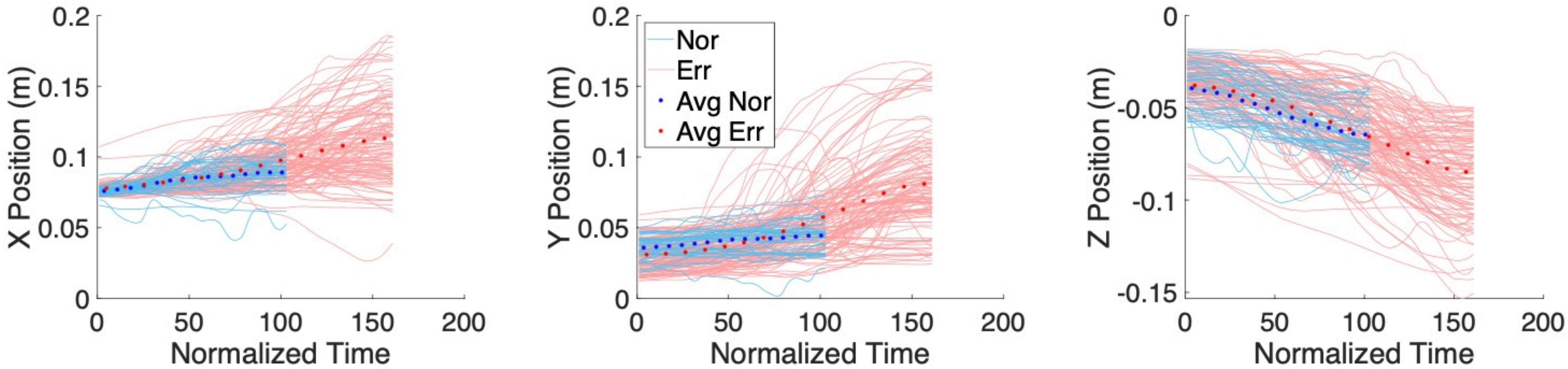
Distribution Similarity Analysis

- Similarity between normal and erroneous trajectories for each gesture measured using Dynamic Time Warping (DTW)
- Kullback-Liebler (KL) divergence between "Err-Nor" and "Nor-Nor" DTW distance distributions

Task	Gesture	Parameters
Suturing	G1	Right Gripper Angle Right Linear Velocity Right Position
	G3	Right Linear Velocity Right Rotational Velocity Right Gripper Angle
	G6	Left Position
	G8	Right Position Left Gripper Angle Left Linear Velocity Right Gripper Angle
	G9	Left Gripper Angle
Needle Passing	G2	Left Rotational Velocity Left Linear Velocity
	G3	Left Rotational Velocity Right Rotation Matrix Right Gripper Angle



Erroneous Gesture Distributions



	SG1	SG2	SG3	SG4	SG6	NPG1	NPG2	NPG3	NPG4	NPG6
SG1	0.00	5.94	4.25	3.04	2.86	**2.77**	4.63	5.26	4.32	4.69
SG2	5.94	0.00	0.70	3.75	3.33	5.82	**0.76**	1.60	4.15	3.66
SG3	4.25	0.70	0.00	3.16	2.36	5.21	1.19	**1.47**	3.21	2.83
SG4	3.04	3.75	3.16	0.00	0.47	3.02	1.76	2.58	**0.49**	0.89
SG6	2.86	3.33	2.36	0.47	0.00	2.11	1.24	1.63	0.48	**0.44**
NPG1	**2.77**	5.82	5.21	3.02	2.11	0.00	2.04	2.27	2.33	2.19
NPG2	4.63	**0.76**	1.19	1.76	1.24	2.04	0.00	0.40	0.83	0.88
NPG3	5.26	1.60	**1.47**	2.58	1.63	2.27	0.40	0.00	1.40	0.95
NPG4	4.32	4.15	3.21	**0.49**	0.48	2.33	0.83	1.40	0.00	0.29
NPG6	4.69	3.66	2.83	0.89	**0.44**	2.19	0.88	0.95	0.29	0.00

Gesture Index	G2	G3	G4	G6
G2	0.00	0.82	0.17	0.73
G3	0.82	0.00	0.82	0.83
G4	0.17	0.82	0.00	0.63
G6	0.73	0.83	0.63	0.00

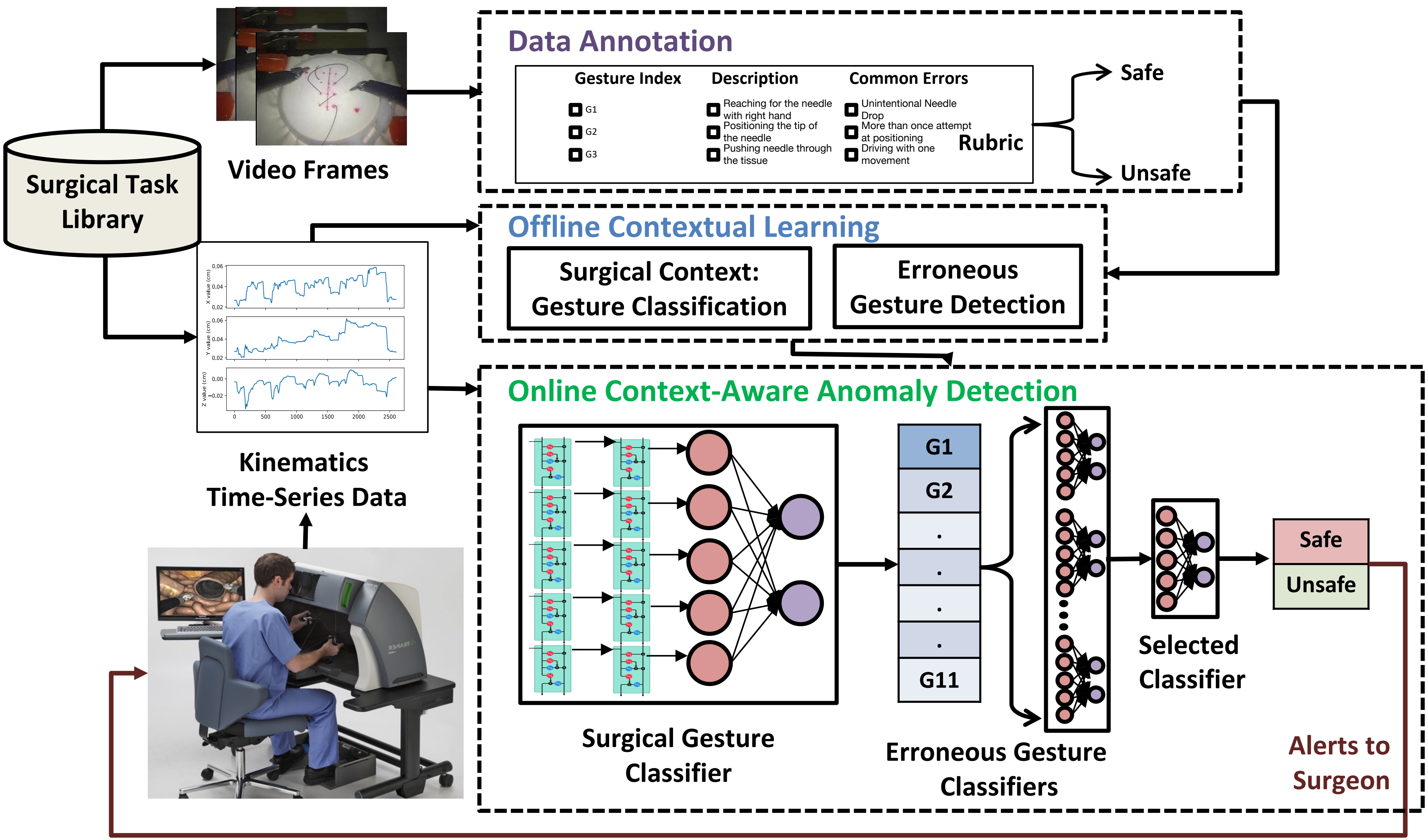
Gesture Index

JS Divergence: 0.0, 0.2, 0.4, 0.6, 0.8

Different error distributions across gesture classes.

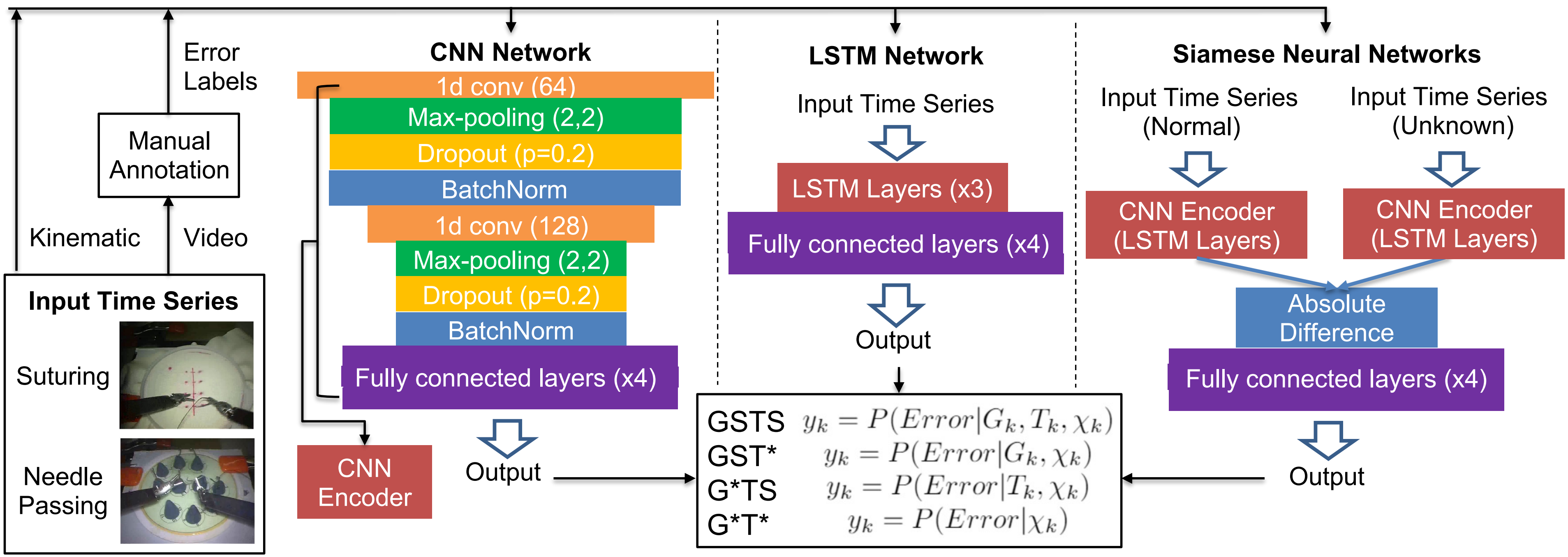
Similarity between the same gestures, even across different tasks.

Context-Aware Runtime Error Detection



Dual NN Architectures for Error Detection

- **Learn to distinguish erroneous from normal trajectories using small data**
  - Leveraging different neural network structures and data grouping techniques



Li, et al., "Runtime Detection of Executional Errors in Robot-Assisted Surgery", Under review.

Context-Specific Data Grouping and Training

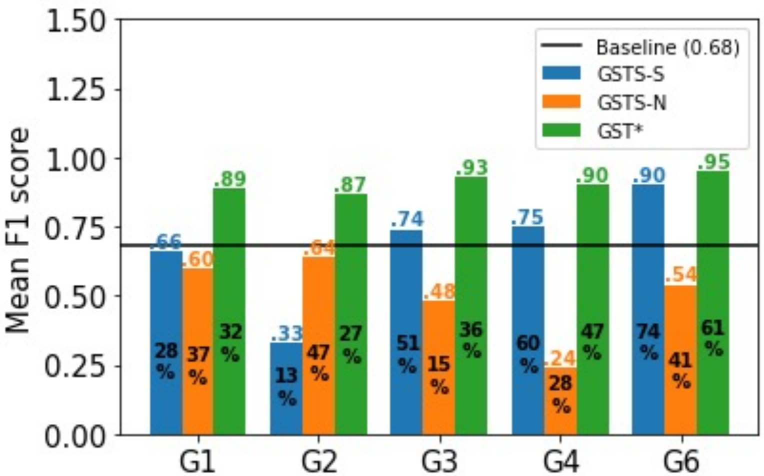
- **Gesture Specific and Task Specific training (GSTS):**
- Separate models were trained for each gesture from each task, resulting in 10 different models.

- **Gesture Specific and Task Nonspecific training (GST*):**
- Five separate models for different gesture classes (G1, G2, G3, G4, G6) by combining gesture-specific data from both tasks.

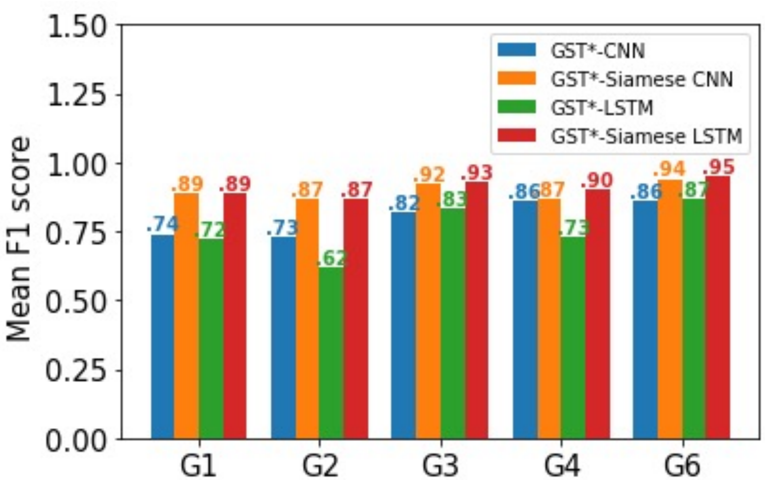
- **Gesture Nonspecific and Task Specific training (G*TS):**
- Two models were trained by combining all different gesture data for each task.

- **Gesture Nonspecific and Task Nonspecific training(G*T*):**
- This setup only included one model as the baseline by training on all the gesture data from both tasks.

Error Detection Performance



Increasing the training data by adding more *relevant* data, such as data from the same gesture class, improves the performance more than adding more irrelevant data.



Dual input Siamese networks and gesture specific models achieve better performance.

Runtime Safety Monitoring

Thanks



UVA Dependable Systems and Analytics Group

NSF
National Institute of Standards and Technology
U.S. Department of Commerce
University of Virginia Engineering
University of Virginia Health System
Commonwealth Cyber Initiative