

# State-Aware Virtual Reality Training for Hydroelectric Maintenance

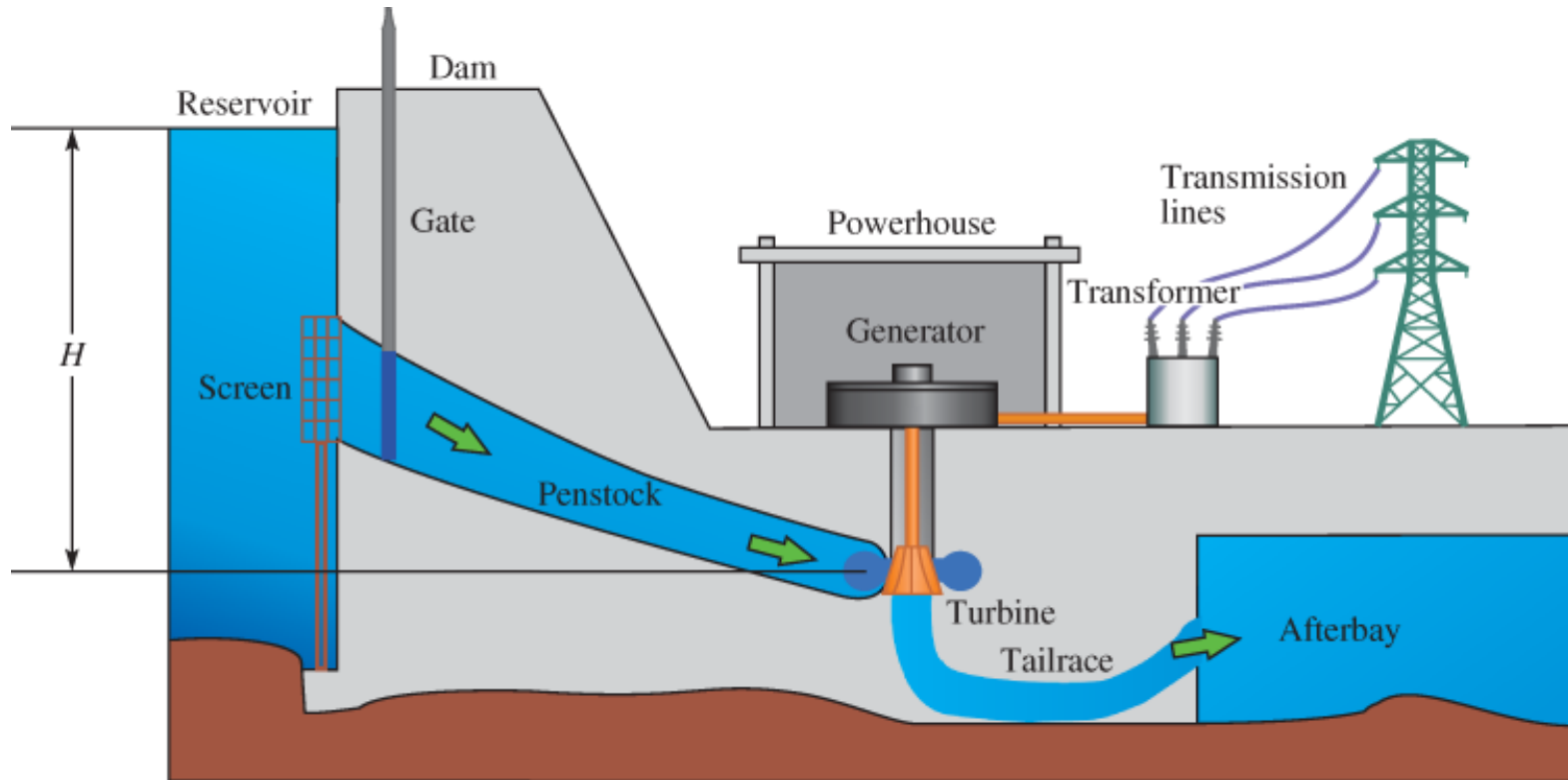
Hatfield, J., Narman H., Chowdhury S., Alzarrad A.

**Joshua Hatfield**

PhD Student in Computer Science, Colorado School of Mines

Master of Science in Computer Science, Marshall University

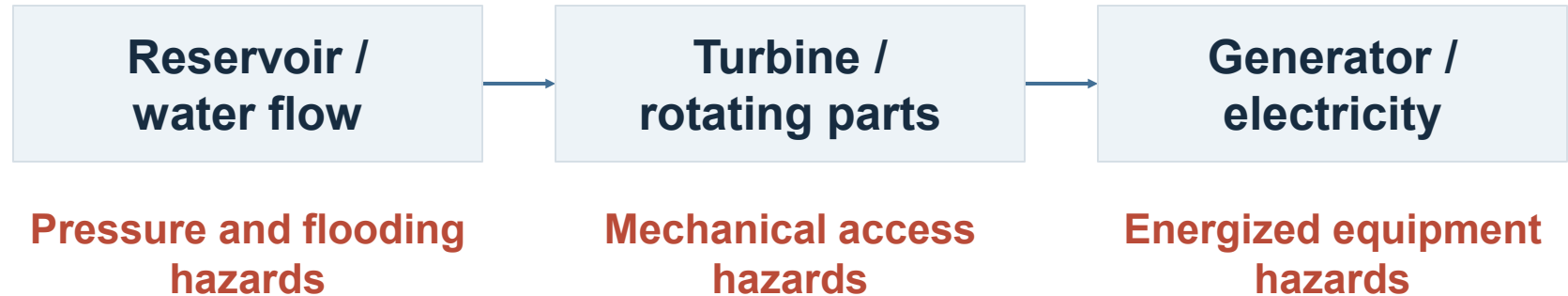
# Introduction: Hydroelectric Plant Overview



Credits to Electrical Academia for Image:

<https://electricalacademia.com/renewable-energy/hydroelectric-power-plant-working-types-hydroelectric-power-plants/>

# Problem: training in a safety-critical environment



**Trainees need to make mistakes without exposure to real plant hazards.**

# Solution: Virtual Reality Training

Fault scenarios can be reproduced without damaging real equipment or exposing trainees to plant hazards.

|                      |                                                          |
|----------------------|----------------------------------------------------------|
| <b>Safe exposure</b> | Practice hazardous scenarios in a simulated environment. |
|----------------------|----------------------------------------------------------|

|                            |                                            |
|----------------------------|--------------------------------------------|
| <b>Repeatable practice</b> | Reset a fault and try the procedure again. |
|----------------------------|--------------------------------------------|

|                             |                                                  |
|-----------------------------|--------------------------------------------------|
| <b>Observable decisions</b> | Capture the actions and timing behind a mistake. |
|-----------------------------|--------------------------------------------------|

**VR provides the setting. State awareness makes the practice interpretable.**

Pratticò et al. [1], Kubr et al. [2], Radhika et al. [12].

# Challenge: Interpreting actions in context

Prior work demonstrates the value of VR for safety-critical training, contextual interaction, and usefulness of state representations

**Procedural state**

What should happen next?

---

**Safety constraints**

Is the action permissible now?

---

**Event telemetry**

What happened, and when?

---

**The framework interprets actions against state and safety constraints.**

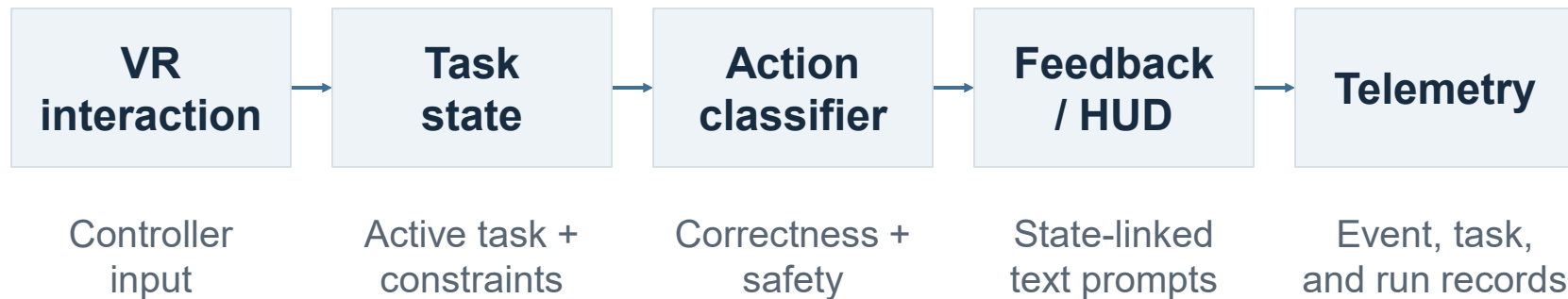
Kosmidis et al. [4] and Bolton et al. [5].

# Goal: Research Questions

- |            |                        |                                                                                                |
|------------|------------------------|------------------------------------------------------------------------------------------------|
| <b>RQ1</b> | <b>Representation</b>  | How to represent state-aware hydroelectric tasks in VR?                                        |
| <b>RQ2</b> | <b>Local behavior</b>  | Does state-aware feedback reduce errors and safety violations compared to no feedback?         |
| <b>RQ3</b> | <b>System outcomes</b> | Do improvements in procedural performance translate to reductions in system-level disruptions? |

# System overview

Each interaction enters the same state-aware assessment loop.

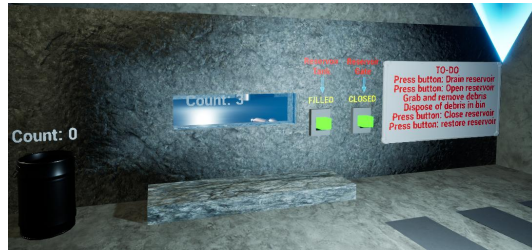


---

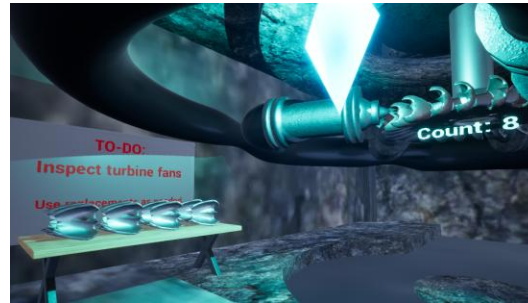
**The classifier links observable behavior to procedural context.**

# The VR maintenance environment

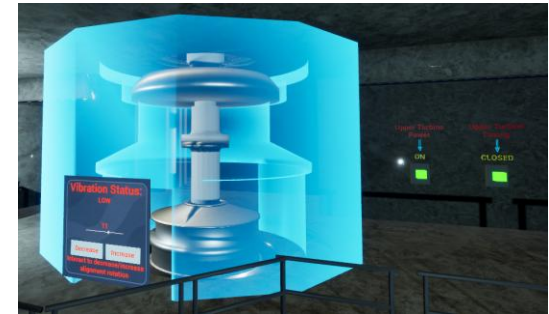
Representative task spaces in one Unreal Engine 5 environment.



Reservoir



Turbine fan

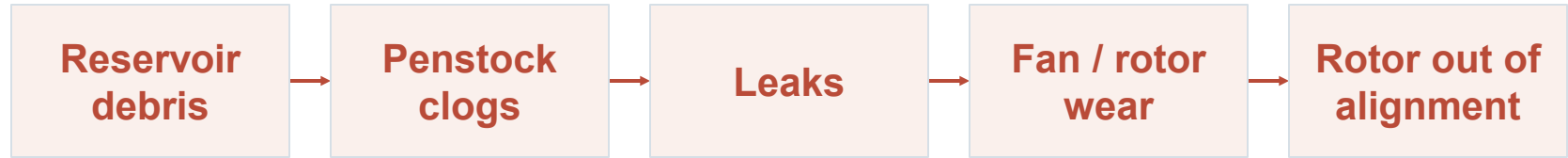


Rotor alignment

These views show task spaces, not separate experimental conditions.

# Cascading faults

A maintenance problem can develop beyond the component where it begins.



**If left unaddressed, a minor issue can propagate into a more severe system disruption**

---

**A correct repair is only part of keeping the whole system stable.**

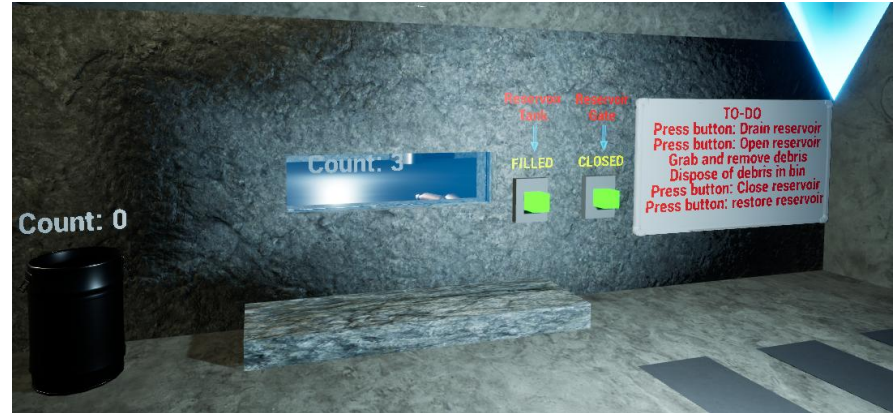
# Task structure

| Task                             | Objective         | Key constraints                  |
|----------------------------------|-------------------|----------------------------------|
| <b>Reservoir Debris Removal</b>  | Remove debris     | Water state and disposal         |
| <b>Penstock Leak Repair</b>      | Patch leaks       | Water shutoff, validate, restart |
| <b>Penstock Clog Removal</b>     | Clear flow path   | Water shutoff and disposal       |
| <b>Turbine fan maintenance</b>   | Replace blades    | Power off, drain, casing, refill |
| <b>Turbine Rotor Maintenance</b> | Replace and align | Power off, casing, alignment     |

# Tasks and Subtasks: Reservoir Example

The reservoir task involves removing debris; it can be decomposed as follows:

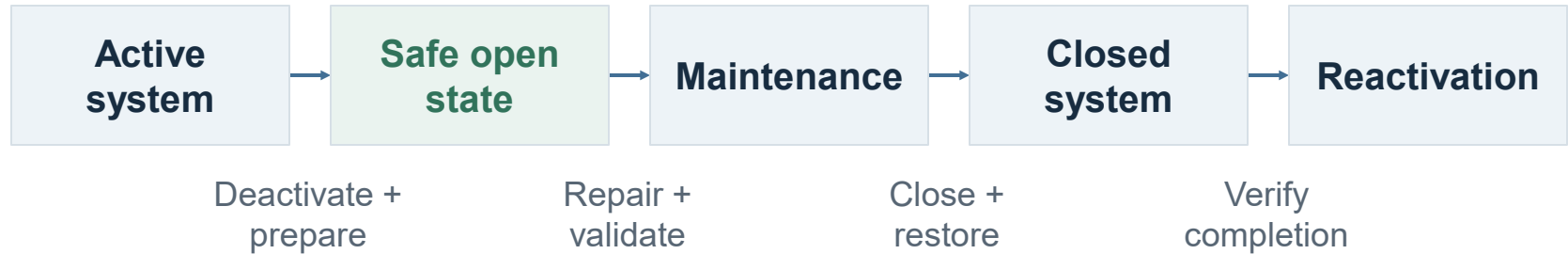
1. Initialization
2. Draining Reservoir Water
3. Opening Reservoir
4. Removing/Disposal of Debris
5. Closing Reservoir
6. Filling Reservoir



RQ1: How to represent state-aware hydroelectric tasks in VR?

# Finite-state procedural model

A simplified common workflow, with task-specific transition guards.



**Example: open the reservoir**   
 → **Reservoir drained: safe access**   
 → **Reservoir filled: safety violation**

The same action changes meaning with the current state.

# Action Classification

State and timing distinguish an action from its interpretation.

|                           |                                                                 |
|---------------------------|-----------------------------------------------------------------|
| <b>Correct action</b>     | Expected step with required constraints satisfied               |
| <b>Wrong-order action</b> | Valid step before its prerequisites                             |
| <b>Safety violation</b>   | Action that breaks an operational constraint                    |
| <b>Hesitation event</b>   | Inactivity beyond the configured threshold                      |
| <b>Event record</b>       | Participant, run, time, subsystem, event, state, classification |

# Feedback condition vs no-feedback condition

RQ2: Does state-aware feedback reduce errors and safety violations compared to no feedback?

## Feedback

HUD text prompts respond to task state, safety constraints, and procedural dependencies.

## No feedback

The same task environment and meters remain visible. No adaptive text prompts.

---

**Implemented feedback: deterministic rules, with offline LLM-assisted authoring.**

# Experiment design

Between-subjects comparison with repeated runs within each condition.

**23**

recruited

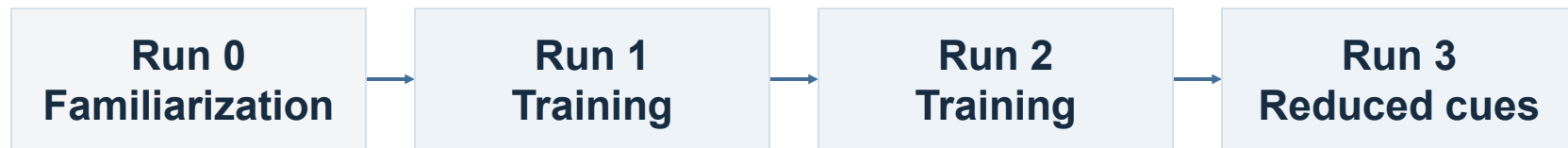
**4**

did not complete

**19**

included

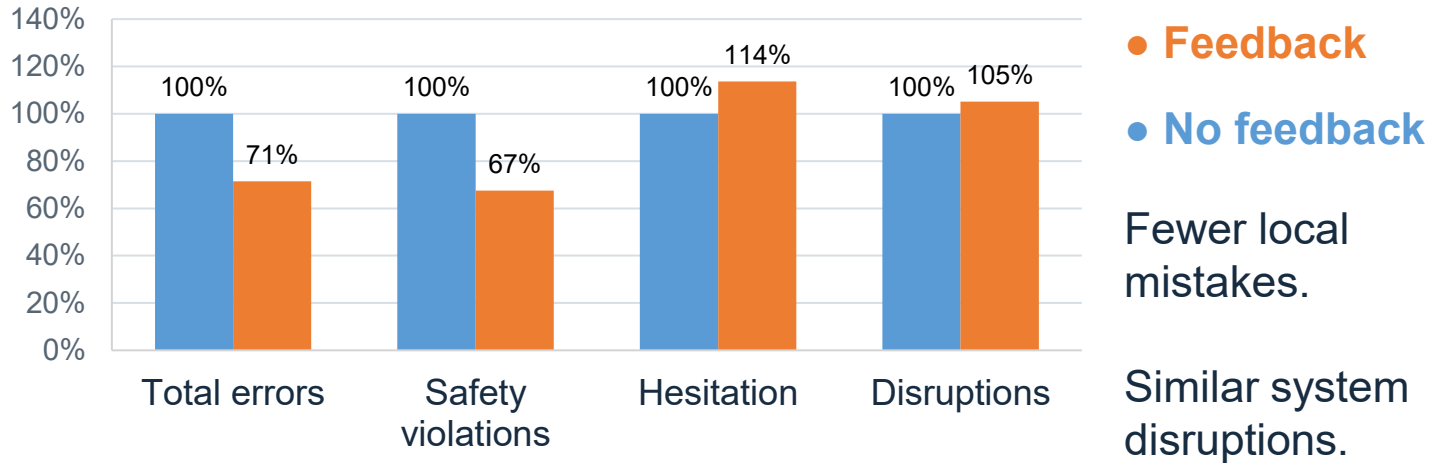
**10 feedback**  
**9 no feedback**



**Repeated practice, then reduced instructional support**

# Local and system results

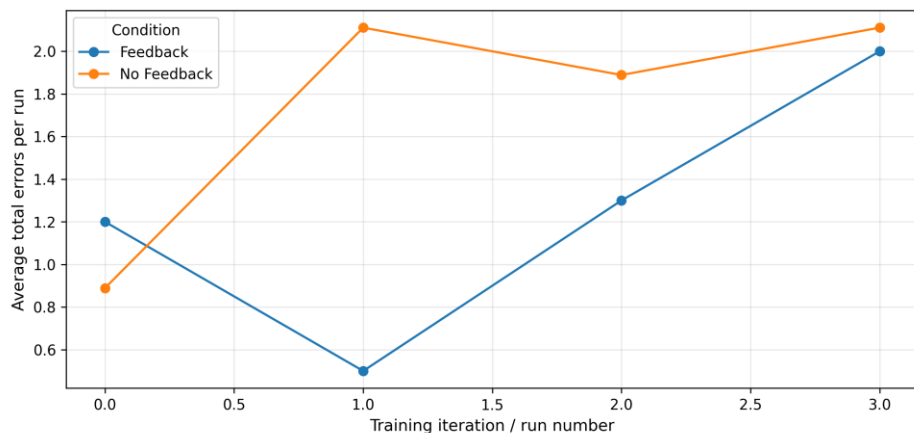
RQ3: Do improvements in procedural performance translate to reductions in system-level disruptions?



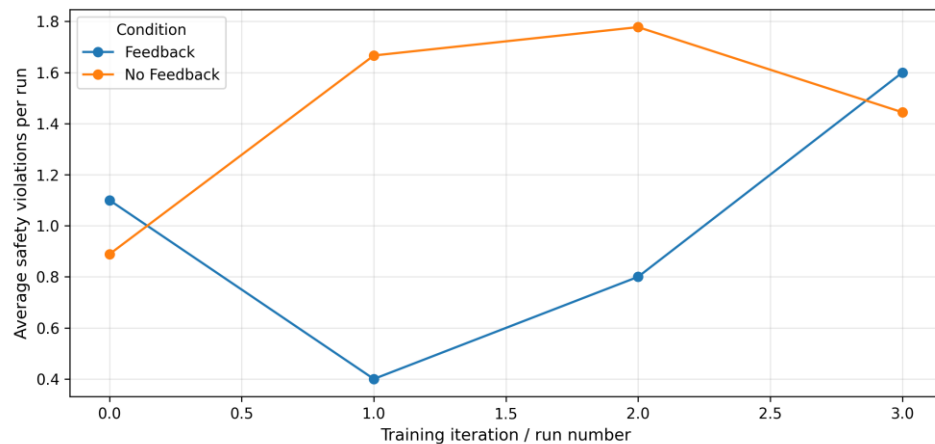
The pattern is promising, but the small study does not establish efficacy.

# Errors and Safety Violations for each Run

Average Total Errors Across Training Iterations



Average Safety Violations Across Training Iterations



Runs indexed at 0

# Interpretation

State-aware prompts appear most useful for local procedural decisions.

## Observed

## Possible explanation

**Fewer local mistakes**

Prompts expose prerequisites and safety constraints.

---

**More hesitation**

Reading, pausing, or reflection may increase delay.

---

**Similar disruptions**

Local guidance may not explain downstream effects.

# Limitations and what they constrain

| Limitation                              | Boundary on the conclusion                  |
|-----------------------------------------|---------------------------------------------|
| <b>Small sample; many comparisons</b>   | Exploratory evidence with limited precision |
| <b>Unvalidated tasks and timers</b>     | Thresholds may shape measured behavior      |
| <b>Deterministic feedback</b>           | No test of a live RL policy                 |
| <b>Normalized system metrics</b>        | Training proxies, not physical equations    |
| <b>Single session; simplified model</b> | Retention and transfer remain unknown       |

# Future work: connecting actions to consequences

|                            |                                                                        |
|----------------------------|------------------------------------------------------------------------|
| <b>Causal feedback</b>     | Explain why local actions affect downstream system state.              |
| <b>Stronger evaluation</b> | Validate tasks and thresholds; model repeated runs with mixed effects. |
| <b>Adaptive policies</b>   | Test live RL policies within explicit safety constraints.              |

---

**VR becomes more useful when trainee actions are interpreted against task state, safety constraints, and system consequences.**

# Research Questions Revisited

## Research Questions

How to represent state-aware hydroelectric tasks in VR?

---

Does state-aware feedback reduce errors and safety violations compared to no feedback?

---

Do improvements in procedural performance translate to reductions in system-level disruptions?

## Observation

Finite-state models link actions to contextually relevant state.

---

Feedback produced marginal improvements to safety violations and errors

---

Better local task performance did not show a corresponding system-level benefit.



# IEEE GHTC 2026

**16th Annual IEEE Global Humanitarian Technology Conference**  
**October 7–10, 2026**  
**University of Colorado Boulder, USA**

**Thank you!**

[Hatfield, J.](#), Narman H., Chowdhury S., Alzarrad A.

# Appendix

TABLE I

PARTICIPANT-LEVEL PERFORMANCE OUTCOMES AGGREGATED ACROSS RUNS 0–3. HEDGES'  $g$  IS CALCULATED AS FEEDBACK MINUS NO FEEDBACK; NEGATIVE VALUES INDICATE LOWER VALUES IN THE FEEDBACK CONDITION.

| Outcome                  | Feedback | No Feedback | $g$    | $p$  |
|--------------------------|----------|-------------|--------|------|
| Total errors             | 5.00     | 7.00        | -0.526 | .254 |
| Wrong-order actions      | 6.70     | 10.56       | -0.753 | .127 |
| Safety violations        | 3.90     | 5.78        | -0.624 | .176 |
| Unnecessary actions      | 12.00    | 19.11       | -0.591 | .222 |
| Mean completion duration | 15.17    | 19.85       | -0.309 | .494 |
| Hesitation count         | 31.30    | 27.56       | 0.683  | .133 |
| Final total disruptions  | 27.90    | 26.56       | 0.093  | .833 |