Show Me What You See: Communicating Visuomotor Policy Attention for Interactive Robot Learning

Jiahe Liu
Department of Computer Science
Indiana University Bloomington
Bloomington, IN, USA
jl456@iu.edu

Michelle Zhao
Department of Computer Science
Indiana University Bloomington
Bloomington, IN, USA
zhaomich@iu.edu

***Abstract*—Interactive imitation learning allows robots to improve through human corrections during deployment, but human teachers typically observe only robot behavior, not the visual representations underlying its policy. We investigate whether these internal visual representations can provide useful feedback during interactive robot learning. Using a recurrent dual-camera visuomotor policy, we analyze SpatialSoftmax features to characterize what task-relevant information is encoded in the policy's visual representation across task stages, full-rollout outcomes, failure windows, and human interventions. We further evaluate camera importance by masking each camera's recent observation history and measuring the resulting change in predicted actions. Predicted actions are more sensitive to wrist-camera history during alignment and insertion. Based on these findings, we instantiate DAgger with visual guidance. After the teacher takes over, the interface provides stage-relevant visual feedback to guide the teacher toward clear, stable observations from task-relevant cameras and support corrective demonstrations. In a within-subjects pilot with two participants, policies fine-tuned on guided-feedback corrections achieved 10–13 percentage-point higher success rates than those fine-tuned on corrections provided with RGB-only camera views. These results provide preliminary evidence that visual policy features can be translated into actionable feedback for human teachers.**

***Index Terms*—human-robot interaction, interactive imitation learning, visuomotor policy, visual attention**

I. Introduction

Offline-trained visuomotor policies can fail in unfamiliar deployment states. Interactive imitation learning lets humans provide corrective demonstrations during deployment [1]–[3], but teachers must still decide *what* the robot should learn from a failure.

This is difficult because policy internals are opaque: similar behavior can arise from attending to the wrong region, missing a task-relevant object, or predicting an incorrect action. Internal visual features may help distinguish these cases, but activation maps are not necessarily causal explanations [4].

We study a dual-camera BC-RNN on the RoboMimic Square task [5]. Using 100 autonomous and 20 human-intervention rollouts, we ask: (1) what task and outcome information is present in SpatialSoftmax features, (2) which cameras, regions, and channels locally affect predicted actions, and (3) how can supported findings provide concise feedback for human correction? Our contributions are:

- a stage-conditioned SpatialSoftmax analysis of rollout outcomes, failure onset, and human correction;
- counterfactual camera, region, and feature-channel ablations of a recurrent visuomotor policy; and
- a human-facing guidance prototype evaluated in an exploratory two-participant pilot.

II. Related Work

Interactive Imitation Learning. Interactive imitation learning (IIL), including DAgger [1], updates robot policies with human feedback collected during deployment [6]. In human-gated methods such as HG-DAgger [7], teachers choose when to intervene; robot-gated methods query teachers based on safety [8], uncertainty [9]–[13], or predicted failure [3], [14], often to reduce intervention demand [11], [15]. These methods focus on *when* to request feedback. We instead study how a visuomotor policy can support teachers during correction, using internal visual information as a feedback source alongside prior nonverbal action feedback [16].

Visual Attention in Robot Visuomotor Policies. Spatial representations, including deep spatial autoencoders and SpatialSoftmax, extract task-relevant feature points for visuomotor control [17]; perturbation-based saliency visualizes image regions that affect decisions [18]. Other work uses human visual attention to improve robot learning, including ViSaRL with human saliency maps [19] and GuidedAttention for inspecting and correcting policy keypoints [20]. In contrast, we use visual information from the robot policy to guide the teacher, exposing SpatialSoftmax findings supported across stages, outcomes, interventions, and camera views.

Explainability and Interpretability in Learning from Demonstration. Feedback about a robot's current understanding can improve human demonstrations [21], but plausible attention maps do not necessarily explain policy actions [4], [22]. Perturbation studies test whether changing inputs changes

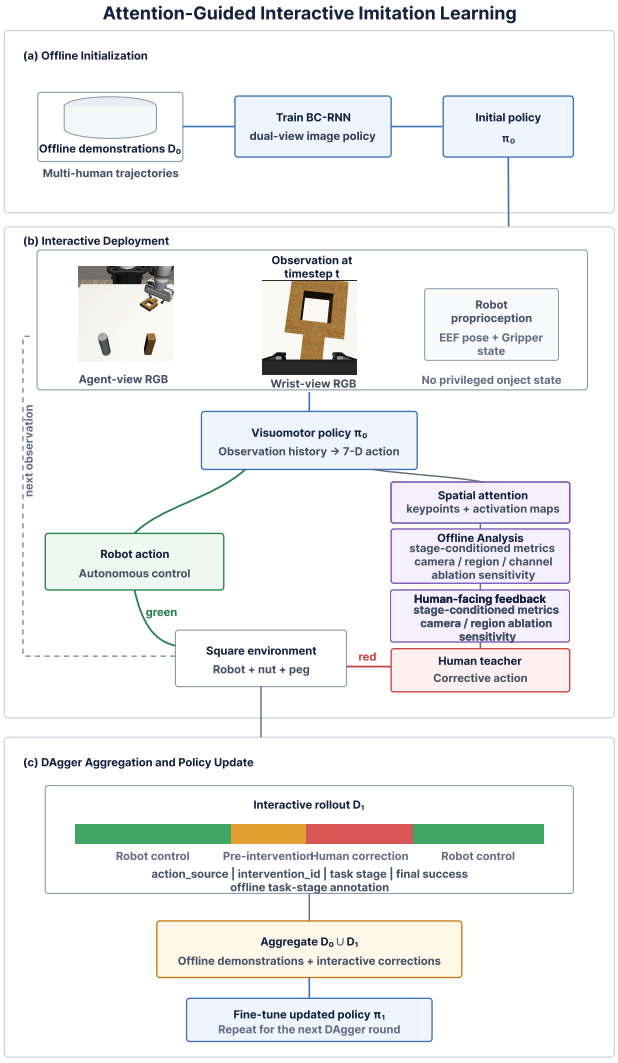


Fig. 1. Overview of visual guidance during interactive robot learning.

outputs [23]. Accordingly, we pair SpatialSoftmax visualizations with camera, region, and feature ablations, presenting visualizations as correction clues and ablation results as local action sensitivity rather than explanations of task success.

III. Pre-Study: Understanding What Can Policy Visual Attention Inform?

Before designing feedback for human teachers, we examine what policy behavior is reflected in its internal visual representation. We conduct a pre-study of SpatialSoftmax features to assess what information they reveal about policy execution and whether it can support human correction.

Specifically, we investigate three questions:

1) **Q1 Rollout outcome:** How do SpatialSoftmax features differ between successful and failed rollouts?
2) **Q2 Task stage:** How do SpatialSoftmax features vary across stages of task execution?
3) **Q3 Camera relevance:** How does the policy's reliance on different camera views vary across task stages?

1) Setup: We use the RoboSuite Square multi-human task [5], where a robot grasps a square nut, transports it to a matching peg, aligns it, and inserts it. The offline dataset D_0 contains 300 demonstrations. Our base policy π_0 is a BC-RNN trained for 460 epochs.

At each timestep, π_0 receives agent-view and wrist-view 84×84 RGB images and 9-D proprioception. Each view is encoded by ResNet-18 [24] and 32-channel SpatialSoftmax [17]. For camera v and channel k, the spatial distribution $P_{t,v,k}(u)$ yields a 2-D keypoint

$$z_{t,v,k} = \sum_{u\in\Omega} P_{t,v,k}(u)c(u), \tag{1}$$

where $c(u)$ is the image coordinate. The 64 keypoints and proprioception form a 137-D feature. A two-layer LSTM processes a 10-step history, and a Gaussian-mixture head predicts a 7-D robot action. Intuitively, $z_{t,v,k}$ indicates where visual channel k focuses more strongly in the current image. It combines the channel's responses across image locations, so regions with stronger responses receive greater weight and pull the resulting 2-D location closer to the visual area the policy emphasizes.

Q1 Rollout Outcome. We run the base policy from 100 distinct saved simulator initial-state snapshots for up to 500 timesteps, obtaining 51 successful and 49 failed rollouts. A single annotator marks failure onset in 41 failed rollouts; the remaining eight are labeled as timeouts. All failed rollouts are included in outcome analysis, while timeout rollouts are excluded only from analyses requiring a failure-start timestep.

To identify which information in the policy's visual representation is associated with rollout outcome, we summarize each camera's SpatialSoftmax maps with 12 scalar metrics at every timestep. These metrics describe where visual features are concentrated and how their spatial distributions and keypoint locations change over time. For example, agent-view lag-10 JS divergence measures the change in each channel's spatial distribution over ten timesteps. For a rollout of length T,

$$s_{\text{agent}} = \frac{1}{32(T-10)} \sum_{t=11}^{T} \sum_{k=1}^{32} \mathrm{JS}(P_{t,\text{agent},k}, P_{t-10,\text{agent},k}), \tag{2}$$

where JS denotes normalized Jensen–Shannon divergence [25]. We summarize each metric over a rollout and use nested five-fold trajectory-level cross-validation to select an outcome-associated descriptor using training rollouts only. The complete metric list appears in Appendix A, and rollout-level evaluation details appear in Appendix B.

Q1 Takeaway: Full-Rollout Metrics Distinguish Outcomes but Not Failure Onset. Table I shows that mean agent-view lag-10 JS divergence best separates completed successful and failed rollouts, with nested AUROC 0.949 ± 0.034; failed rollouts generally show less temporal change. However, successful

TABLE I
MAIN REPRESENTATION-ANALYSIS RESULTS.

Question	Metric	Result
Full rollout outcome	AUROC	0.949 ± 0.034
Fixed 150-step outcome	AUROC	0.691 ± 0.021
Task stage	Balanced acc.	0.654
Task stage	Macro-F1	0.600
Early baseline vs. pre-failure	AUROC	0.663
Pre- vs. post-failure	AUROC	0.627

rollouts often terminate earlier, while failures reach the 500-step horizon. Restricting every rollout to the first 150 steps reduces AUROC to 0.691 ± 0.021, showing that full-rollout association includes information from later observations and trajectory duration. The weaker onset-based analyses in Table I further show that these metrics do not reliably identify the precise onset of failure online.

Q2 Task Stage. We ask whether SpatialSoftmax metrics vary across normal task stages. Using the 100 autonomous pre-study rollouts, we assign offline labels for Approach, Grasp, Transport/Align, and Insert from simulator geometry. These labels are used only for analysis and are not available to the policy. At each timestep, we use the 12 SpatialSoftmax metrics from both cameras as input to a class-balanced multinomial logistic regression model. We sample at most 50 timesteps per stage from each trajectory and use trajectory-grouped five-fold cross-validation.

Q2 Takeaway: SpatialSoftmax Metrics Contain Task-Stage Information. The classifier achieves accuracy 0.620, balanced accuracy 0.654, and macro-F1 0.600. Thus, SpatialSoftmax metrics contain information about task progress, but are insufficient for high-reliability online stage detection.

Q3 Camera Relevance. We measure each camera's local influence on predicted actions using counterfactual masking. For sampled autonomous 10-step observation histories, we replace agent-view or wrist-view RGB inputs with view-specific mean RGB values while keeping proprioception unchanged. We compare the normalized expected actions $a_t(A, W)$, $a_t(A, 0)$, $a_t(0, W)$, and $a_t(0, 0)$, masking either the current frame or the full history. To account for camera interactions, we compute two-player Shapley-style contributions:

$$\begin{aligned} \phi_A &= \frac{1}{2}\left[a_t(A,0) - a_t(0,0) + a_t(A,W) - a_t(0,W)\right], \\ \phi_W &= \frac{1}{2}\left[a_t(0,W) - a_t(0,0) + a_t(A,W) - a_t(A,0)\right]. \end{aligned} \tag{3}$$

For camera $v \in \{A, W\}$, we compute its RMS action effect and normalized sensitivity share:

$$I_v = \sqrt{\frac{1}{7}\sum_{d=1}^{7} \phi_{v,d}^2}, \qquad S_v = \frac{I_v}{I_A + I_W} \times 100\%. \tag{4}$$

These shares measure local action sensitivity under mean-RGB masking, rather than attention weights, confidence, or causal effects on task success.

TABLE II
MEDIAN SENSITIVITY SHARE UNDER 10-STEP CAMERA-HISTORY MASKING.

Stage	Agent	Wrist
Approach	43.2%	56.8%
Grasp	42.5%	57.5%
Transport/Align	19.8%	80.2%
Insert	12.4%	87.6%

Q3 Takeaway: Wrist-History Sensitivity Increases in Precision Stages. Current-frame masking yields similar agent and wrist sensitivity, 52.5% and 47.5%. Because the LSTM uses observation history, we also mask each camera across the preceding 10 steps. Wrist sensitivity then accounts for 68.5% overall, increasing during precision stages as shown in Table II. We therefore recommend the wrist view during Transport/Align and Insert; these percentages indicate ablation sensitivity, not confidence.

Changes During Human Intervention. We analyze 20 successful human-gated DAgger rollouts with 22 intervention segments, comparing wrist-view features before takeover, during human control, and after policy control resumes. Nineteen Grasp interventions occur across 17 rollouts. During human control, JS divergence and keypoint movement increase in all 17 rollouts, with median standardized changes of $+1.640$ and $+1.123$ and BH $q < 0.001$. Both measures decrease after policy control resumes in 88% of rollouts, with median changes of -1.429 and -0.865 and $q = 0.002$. These transient changes do not explain failure causes or correction quality.

IV. METHODOLOGY: INTERACTIVE TEACHING WITH VISUAL GUIDANCE

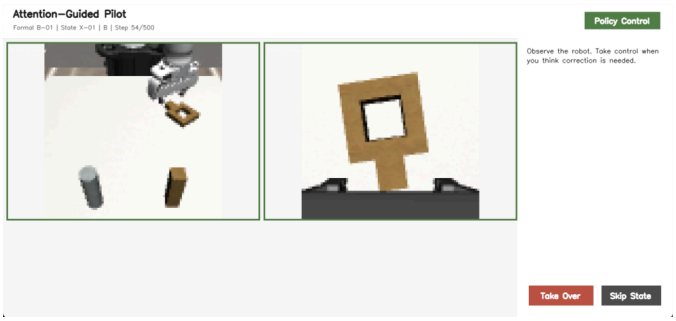
We translate the supported pre-study findings into concise visual guidance for human correction. Figure 1 summarizes the guidance-generation procedure.

A. Stage-Specific Visual Guidance

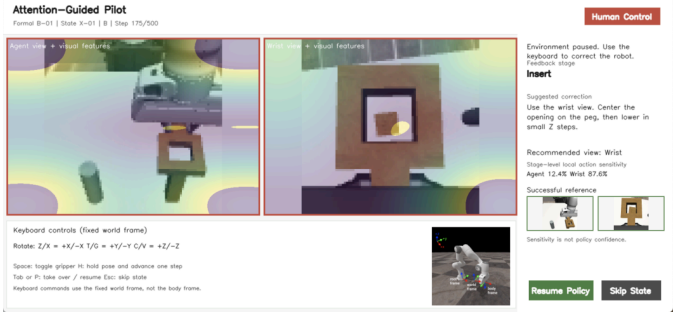
During policy control, the teacher observes agent-view and wrist-view RGB images and decides when to take over. We do not display an automatic failure warning because local SpatialSoftmax metrics do not reliably identify failure onset.

After takeover, a separate gated recurrent unit (GRU) stage detector summarizes policy-available observation and action history to predict Approach/Grasp, Transport/Align, or Insert. The predicted stage selects a prepared profile with a SpatialSoftmax heatmap, recommended camera view, successful same-stage reference, and correction cue.

The camera recommendation is derived from 10-step history masking; the wrist view is recommended during Transport/Align and Insert. The heatmap is initialized at takeover and refreshed approximately every five actions during uninterrupted human control. References and cues are manually prepared, and camera sensitivity is presented as local action sensitivity rather than confidence or a causal explanation. Figure 2 shows the interface.



(a) RGB-only observations



(b) Guided feedback after takeover

Fig. 2. Teaching interface used in the pilot. (a) RGB-only observations shown in Condition A and during policy control in Condition B; participants observe the agent-view and wrist-view RGB images and decide when to take over. (b) Additional guidance shown after takeover in Condition B, including SpatialSoftmax overlays, the predicted feedback stage, a recommended camera view, a successful same-stage reference, and a correction cue.

V. Pilot Study: Is Policy-Derived Guidance Helpful?

Study Design. We conduct an exploratory within-subjects pilot with two participants. In both conditions, participants observe agent-view and wrist-view RGB images and decide when to take over. Condition A provides RGB-only feedback in Fig. 2(a), while Condition B adds guided feedback after takeover in Fig. 2(b). Participant 1 completes A on set X and B on set Y; Participant 2 completes the reverse order.

Fine-Tuning and Evaluation. For each participant-condition pair, we fine-tune the common epoch-460 BC-RNN checkpoint for 200 epochs on D_0 plus six full mixed-control rollouts. Full rollouts retain states before, during, and after correction. We evaluate all four fine-tuned policies and the original checkpoint at the fixed epoch 200 on the same 100 initial states, disjoint from the pre-study and pilot sets.

Results. Table III summarizes the retained data and fixed-state evaluation. In the initial evaluation, guided-feedback fine-tunes achieve higher success than the corresponding RGB-only fine-tunes: 68% versus 55% for P1 and 59% versus 49% for P2. The original epoch-460 policy achieves 50%. These results provide exploratory end-to-end evidence that policy-derived guidance can collect useful corrective data, but the pilot is too small and selectively retains successful trajectories; it does not establish a causal effect of guided feedback.

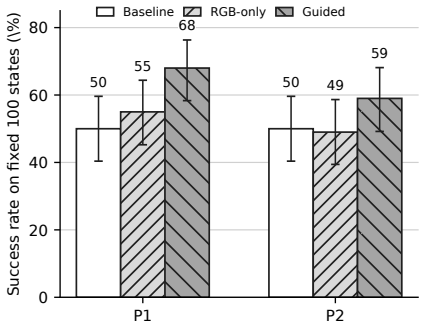


Fig. 3. Initial evaluation on 100 unseen fixed states. Each policy is fine-tuned from the common BC-RNN checkpoint using D_0 and six mixed-control rollouts. Error bars show 95% Wilson intervals.

TABLE III
Pilot collection and initial downstream evaluation.

Participant	Condition	Intv.	Human steps (%)	Success
P1	RGB-only	8	307 (21.2)	55%
P1	Guided	20	315 (16.2)	68%
P2	RGB-only	11	589 (26.5)	49%
P2	Guided	8	180 (10.4)	59%

VI. Discussion

Full-rollout SpatialSoftmax metrics distinguish successful and failed trajectories, but fixed-prefix and failure-window results show that they do not reliably identify failure onset online. Camera-history masking provides the clearest actionable result: predicted actions are more sensitive to wrist-camera history during Transport/Align and Insert. Region and channel analyses remain useful for inspection, but do not provide causal explanations of task outcome.

This study uses one simulated task and one BC-RNN policy. Offline stage labels rely on simulator geometry, failure starts are marked by one annotator, and the pilot includes only two participants. Because the pilot retains only successful trajectories with human takeover, its fine-tuning results are descriptive and cannot estimate trial-level effects of guidance. Successful references and correction cues are also manually prepared.

VII. Conclusion

We use SpatialSoftmax analysis and camera-history masking to build a human-facing guidance interface with stage-relevant camera recommendations, successful references, and correction cues. In the initial fixed-state pilot evaluation, guided-feedback data produces higher downstream success than RGB-only data for both participant-specific fine-tunes. Future work will collect human gaze data, study its alignment with policy visual representations, and explore whether it can provide an additional visual training signal for the policy. Larger controlled studies are needed to evaluate intervention behavior, correction quality, and policy learning across participants, tasks, and robots.

APPENDICES

APPENDIX A
POLICY ARCHITECTURE AND SPATIALSOFTMAX FEATURES

Each camera image is encoded by ResNet-18 and a 32-channel SpatialSoftmax layer. For camera v, channel k, and image location $u \in \Omega$, the SpatialSoftmax distribution $P_{t,v,k}(u)$ produces a 2-D keypoint

$$z_{t,v,k} = \sum_{u \in \Omega} P_{t,v,k}(u) c(u), \tag{5}$$

where $c(u)$ is the normalized image coordinate. The 64 keypoints from two cameras are concatenated with 9-D proprioception into a 137-D feature. A two-layer LSTM processes a 10-step history, and a Gaussian-mixture head predicts a 7-D action.

At each timestep and for each camera, we compute 12 SpatialSoftmax metrics: normalized entropy, spatial spread, peak probability, keypoint dispersion, one-step keypoint movement, one-step JS divergence, 90th-percentile movement across channels, 90th-percentile JS divergence across channels, lag-10 keypoint displacement, lag-10 JS divergence, rolling-20 keypoint activity, and rolling-20 JS activity. The rolling metrics are trailing averages over up to 20 timesteps.

APPENDIX B
ROLLOUT-OUTCOME METRICS AND EVALUATION

For distributions P and Q, normalized Jensen–Shannon divergence is

$$\mathrm{JS}(P, Q) = \frac{\mathrm{KL}(P\|M) + \mathrm{KL}(Q\|M)}{2 \log 2}, \qquad M = \frac{P+Q}{2}. \tag{6}$$

For example, lag-10 JS divergence compares each channel's SpatialSoftmax distribution at timestep t with that at $t-10$.

For every rollout, we summarize each of the 12 metrics from each camera using its mean, median, and 95th percentile across valid timesteps. This produces $12 \times 2 \times 3 = 72$ rollout descriptors. We use five trajectory-level folds stratified by final success. Within each fold, we select the descriptor and score direction with the strongest training-fold association with failure, then calculate AUROC on held-out trajectories. Reported full-rollout AUROC is the mean and standard deviation across the five held-out folds.

We repeat the same nested procedure using the first 150 timesteps of every rollout. This is a common prefix available for all rollouts, including the shortest successful rollout of 152 steps. The fixed-prefix analysis controls for different trajectory durations and reduces the influence of later post-failure behavior.

For failure-window analyses, each timestep is represented by 24 features: the same 12 SpatialSoftmax metrics computed separately for the agent and wrist cameras. These features describe how concentrated each camera's SpatialSoftmax distributions are and how their keypoints and spatial distributions change over time. We use median imputation, feature standardization, class-balanced logistic regression, and trajectory-grouped five-fold cross-validation. For an annotated failure start f, the early-warning comparison uses $[f-60, f-40)$ as the early baseline and $[f-20, f)$ as the pre-failure window. The pre/post comparison uses $[f-20, f)$ and $[f, f+20)$. Timesteps outside these windows are excluded. We pool held-out predictions from all folds to calculate AUROC.

APPENDIX C
TASK-STAGE, FAILURE-WINDOW, AND INTERVENTION ANALYSES

Offline task labels assign timesteps to Approach, Grasp, Transport/Align, or Insert using simulator geometry and reviewed stage boundaries. These labels are used only for analysis. The stage classifier receives the 24 SpatialSoftmax metrics, samples at most 50 timesteps per stage from each trajectory, and uses class-balanced multinomial logistic regression with trajectory-grouped five-fold cross-validation.

For intervention analysis, we use 20 successful human-gated DAgger rollouts containing 22 interventions. We compare same-stage wrist-view metrics before takeover, during human control, and after policy control resumes. Phase values are summarized by their median. Repeated interventions are collapsed by rollout and takeover stage. The delayed post-intervention window covers steps 21–35 after human control ends.

The research dashboard supports synchronized offline inspection of camera observations, SpatialSoftmax maps and keypoints, temporal metrics, task stages, and failure or intervention annotations.

APPENDIX D
COUNTERFACTUAL VISUAL ABLATIONS

Camera, region, and channel analyses use sampled 10-step histories from autonomous rollouts. These histories include reference points from the four normal task stages, pre-onset failure points, and matched successful controls. Proprioception remains unchanged in all ablations.

For camera masking, agent-view and wrist-view RGB observations are replaced with their corresponding mean RGB values. We evaluate the normalized expected actions $a_t(A, W)$, $a_t(A, 0)$, $a_t(0, W)$, and $a_t(0, 0)$, masking either the current frame or all ten history frames. Camera contributions are

$$\phi_A = \frac{1}{2}\left[a_t(A,0) - a_t(0,0) + a_t(A,W) - a_t(0,W)\right], \tag{7}$$

$$\phi_W = \frac{1}{2}\left[a_t(0,W) - a_t(0,0) + a_t(A,W) - a_t(A,0)\right]. \tag{8}$$

For $v \in \{A, W\}$, we calculate RMS action effect and normalized sensitivity share as

$$I_v = \sqrt{\frac{1}{7}\sum_{d=1}^{7} \phi_{v,d}^2}, \qquad S_v = \frac{I_v}{I_A + I_W} \times 100\%. \tag{9}$$

For region masking, we divide the central 76×76 pixels of the current 84×84 image into a 3×3 grid. SpatialSoftmax

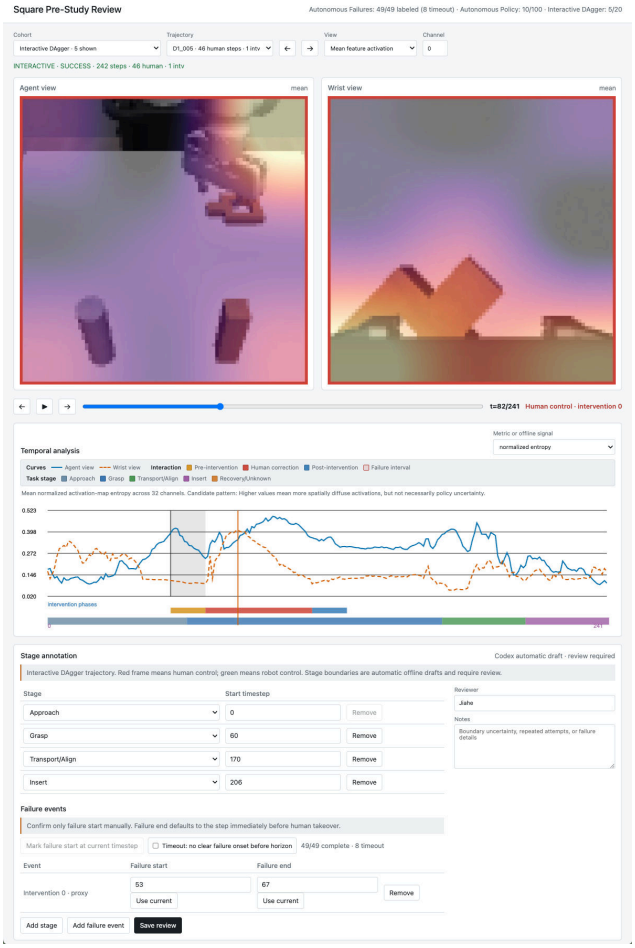


Fig. 4. Interfaces used in our study. The research-facing dashboard supports synchronized offline inspection of camera observations, temporal metrics, task stages, and failure or intervention annotations.

maps are combined using channel-sensitivity weights. We mask the highest-scoring cell and compare its action effect with the lowest-scoring and deterministic random cells. Only the current frame is masked.

For channel replacement, each of the 64 SpatialSoftmax keypoints is replaced throughout the 10-step history by its dataset-wide mean coordinate $\bar{z}_{v,k}$. The channel effect is

$$C_{v,k,t} = \sqrt{\frac{1}{7}\sum_{d=1}^{7}\left(a_{t,d}^{(z_{v,k}\leftarrow\bar{z}_{v,k})} - a_{t,d}\right)^2}. \qquad (10)$$

Channels are ranked by mean action change across sampled histories. All ablation values measure local action sensitivity under the selected intervention baseline, not policy confidence or causal effects on task success.

Appendix E
Pilot Retention and Descriptive Statistics

A formal pilot rollout is retained only if it contains a human takeover, at least one human-action timestep, and final task success. Each participant contributes six retained mixed-control rollouts per condition. Skipped trials and trials without effective human correction are not recorded; trials that fail after takeover are retried from the same initial state. Therefore, these statistics describe retained corrective data rather than trial-level success or takeover rates.

References

[1] S. Ross, G. Gordon, and D. Bagnell, "A reduction of imitation learning and structured prediction to no-regret online learning," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 627–635.

[2] A. Mandlekar, D. Xu, R. Martín-Martín, Y. Zhu, L. Fei-Fei, and S. Savarese, "Human-in-the-loop imitation learning using remote teleoperation," *arXiv preprint arXiv:2012.06733*, 2020.

[3] H. Liu, S. Nasiriany, L. Zhang, Z. Bao, and Y. Zhu, "Robot learning on the job: Human-in-the-loop autonomy and learning during deployment," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[4] S. Jain and B. C. Wallace, "Attention is not explanation," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 3543–3556.

[5] A. Mandlekar, D. Xu, J. Wong, S. Nasiriany, C. Wang, R. Kulkarni, L. Fei-Fei, S. Savarese, Y. Zhu, and R. Martín-Martín, "What matters in learning from offline human demonstrations for robot manipulation," *arXiv preprint arXiv:2108.03298*, 2021.

[6] M. Zare, P. M. Kebria, A. Khosravi, and S. Nahavandi, "A survey of imitation learning: Algorithms, recent developments, and challenges," *IEEE Transactions on Cybernetics*, vol. 54, no. 12, pp. 7173–7186, 2024.

[7] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer, "Hg-dagger: Interactive imitation learning with human experts," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 8077–8083.

[8] J. Zhang and K. Cho, "Query-efficient imitation learning for end-to-end autonomous driving," *arXiv preprint arXiv:1605.06450*, 2016.

[9] K. Menda, K. Driggs-Campbell, and M. J. Kochenderfer, "Ensembledagger: A bayesian approach to safe imitation learning," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 5041–5048.

[10] M. Zhao, H. Admoni, R. Simmons, A. Ramdas, and A. Bajcsy, "Conformalized interactive imitation learning: Handling expert shift and intermittent feedback," in *International Conference on Learning Representations*, vol. 2025, 2025, pp. 9238–9251.

[11] R. Hoque, A. Balakrishna, C. Putterman, M. Luo, D. S. Brown, D. Seita, B. Thananjeyan, E. Novoseller, and K. Goldberg, "Lazydagger: Reducing context switching in interactive imitation learning," in *2021 IEEE 17th international conference on automation science and engineering (case)*. IEEE, 2021, pp. 502–509.

[12] S.-W. Lee, X. Kang, and Y.-L. Kuo, "Diff-dagger: Uncertainty estimation with diffusion policy for robotic manipulation," in *2025 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2025, pp. 4845–4852.

[13] R. Hoque, L. Y. Chen, S. Sharma, K. Dharmarajan, B. Thananjeyan, P. Abbeel, and K. Goldberg, "Fleet-dagger: Interactive robot fleet learning with scalable human supervision," in *Conference on Robot Learning*. PMLR, 2023, pp. 368–380.

[14] H. Liu, Y. Zhang, V. Betala, E. Zhang, J. Liu, C. Ding, and Y. Zhu, "Multi-task interactive robot fleet learning with visual world models," *arXiv preprint arXiv:2410.22689*, 2024.

[15] R. Hoque, A. Balakrishna, E. Novoseller, A. Wilcox, D. S. Brown, and K. Goldberg, "Thriftydagger: Budget-aware novelty and risk gating for interactive imitation learning," *arXiv preprint arXiv:2109.08273*, 2021.

[16] S. H. Huang, I. Huang, R. Pandya, and A. D. Dragan, "Nonverbal robot feedback for human teachers," *arXiv preprint arXiv:1911.02320*, 2019.

[17] C. Finn, X. Y. Tan, Y. Duan, T. Darrell, S. Levine, and P. Abbeel, "Deep spatial autoencoders for visuomotor learning," in *2016 IEEE International conference on robotics and automation (ICRA)*. IEEE, 2016, pp. 512–519.

[18] S. Greydanus, A. Koul, J. Dodge, and A. Fern, "Visualizing and understanding atari agents," in *International conference on machine learning*. PMLR, 2018, pp. 1792–1801.

[19] A. Liang, J. Thomason, and E. Bıyık, "Visarl: Visual reinforcement learning guided by human saliency," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2024, pp. 2907–2912.

[20] M. Murooka, R. Nakajo, K. Shirai, T. Motoda, H. Oh, R. Hanai, and Y. Domae, "Guidedattention: Interpretable and correctable visual attention for ood-robust robot manipulation via imitation learning," 2026. [Online]. Available: https://arxiv.org/abs/2607.21049

[21] S. H. Huang, I. Huang, R. Pandya, and A. D. Dragan, "Nonverbal robot feedback for human teachers," in *Proceedings of the Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, vol. 100, 2020, pp. 1038–1051. [Online]. Available: https://proceedings.mlr.press/v100/huang20a.html

[22] J. Adebayo, J. Gilmer, M. Muelly, I. Goodfellow, M. Hardt, and B. Kim, "Sanity checks for saliency maps," in *Advances in Neural Information Processing Systems*, vol. 31, 2018. [Online]. Available: https://papers.neurips.cc/paper_files/paper/2018/hash/294a8ed24b1ad22ec2e7efea049b8737-Abstract.html

[23] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*, 2014, pp. 818–833. [Online]. Available: https://arxiv.org/abs/1311.2901

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[25] J. Lin, "Divergence measures based on the shannon entropy," *IEEE Transactions on Information theory*, vol. 37, no. 1, pp. 145–151, 1991.