

Instrumentation for Imitation Learning: Enhancing Training Datasets for Clothes Hanger Insertion

Remko Proesmans¹, Thomas Lips¹ and Francis wyffels¹

Abstract—Large behaviour models have transformed the field of robotic manipulation, but prohibitive data requirements have thus far prevented a revolution similar to vision language models. We believe that instrumentation, i.e. sensor integration in objects, can provide invaluable state information and enable efficient learning for robotic manipulation. In this paper, we present instrumented imitation learning of clothes hanger insertion. Using 180 teleoperated demonstrations, we train diffusion policies with and without access to instrumentation data. Results show that policies leveraging instrumentation outperform vision-only counterparts by 14–25 %pt and exhibit greater task awareness. Crucially, a black-box imitation learning policy learns to prioritise instrumentation signals without explicit guidance. In addition, enhancing the teleoperation dataset with rollouts from an instrumented “expert” policy, enables a vision-only “student” policy to achieve performance comparable to the instrumented expert, thereby surpassing the original vision-only policy. These findings establish instrumentation as a promising strategy to enhance imitation learning for robotic manipulation. Datasets are available on Zenodo [[10.5281/zenodo.17122216](https://zenodo.org/record/17122216)].

I. INTRODUCTION

Great expectations currently surround the development of large behaviour models (LBMs) for robotics because of the recent success of vision-language models (VLMs) in image and text generation, or software-based environments in general [1]–[3]. However, it remains to be seen whether these techniques can transfer to real-time sensing, planning, and control for physical machines operating in unpredictable, human environments [1]. At the very least, we are far from reaching the internet-scale data requirements to make them feasible [2]. A popular way to achieve advanced manipulation skills for robots is to make them imitate human behaviour through demonstrations, called imitation learning (IL) [4]. This has proven to be effective in teaching complex manipulation skills to robots [5]–[8]. However, learning even a single task in a controlled environment can take 100 or more demonstrations [5], [6]. For a robot to handle multiple complex tasks in unstructured environments, the data requirements grow tremendously.

Large datasets for robot pre-training exist [9]–[11], but current LBMs still require task-specific fine-tuning for complex tasks or diverse environments [7]. Obtaining larger datasets can be done by adding successful policy rollouts to the training set and retraining, but this merely biases the data

*This work was supported by the Research Foundation Flanders (FWO) under grant agreement no. 1S15925N and 1S56024N, and by the euROBIN Project (EU grant number 101070596).

¹Remko Proesmans, Thomas Lips and Francis wyffels are with the AI and Robotics Lab (IDLab-AIRO), Ghent University—imec, Ghent, Belgium remko.proesmans@ugent.be

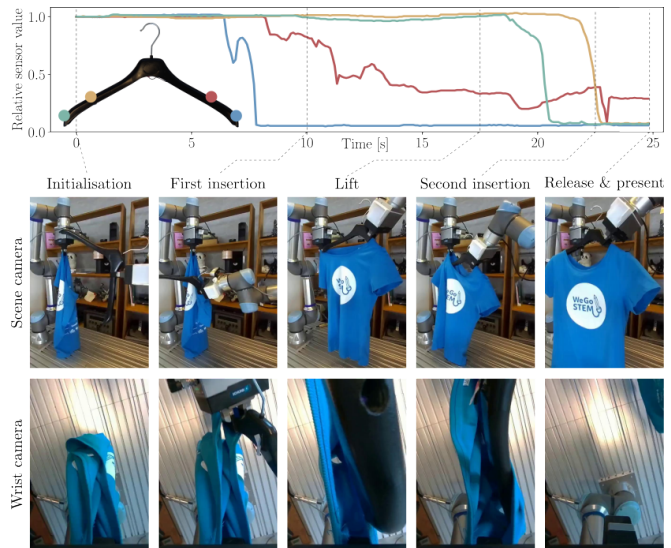


Fig. 1: Instrumented clothes hanger insertion.

distribution toward states where the policy is already proficient [12]. Alternatively, in interactive imitation learning [13] expert interventions can get a struggling policy rollout back on track [14]–[16]. This allows for deliberate data collection to efficiently overcome data mismatches between the train data and deployment settings [15].

In addition to larger datasets and algorithmic advances, we can attempt to improve the quality of the demonstrations to increase IL capabilities. Both [17] and [18] determined evaluation metrics for the quality of demonstration trajectories, and showed that better demonstrations can result in a better policy for the same amount of data. Xu et al. [19] have shown that reinforcement learning (RL) agents generate high-quality trajectories through reward maximisation, making them better suited for fine-tuning generalist policies compared to human demonstrations.

In [20], a different approach from standard IL was used for raspberry harvesting: instead of teleoperating the robot and recording trajectories as demonstrations, human demonstrators manipulated an instrumented, i.e. sensorised, strawberry phantom. Then, the parameters of a custom controller were automatically tuned to match human behaviours as experienced by the strawberry. Instrumentation was also used in [21] for learning to solve a Rubik’s cube with an anthropomorphic robot hand. Specifically, the sensor data provided accurate face angle observations of the Rubik’s cube, adding to the state information needed to shape the

reward function. However, when moving to a vision-only policy, the sensorised cube was discarded and replaced by a synthetically trained vision model: the sensorised cube was rather a stepping stone in development, than a critical influence in obtaining a functional policy. Similarly, [22] and [23] use sensors for state estimation of cloth to enable robotic folding, but do not yet transfer to garments without sensors.

The fundamental idea of instrumentation is to use additional state information during the learning phase and then deploy without this privileged information, using only sensory input that is available in the field. This concept is also found in works that learn robot behaviours in simulation [24], [25]. In [24], sim-to-real quadrupedal locomotion is achieved by first training a teacher policy that has access to privileged state information of the robot, allowing it to quickly achieve high performance. The teacher is then used to guide the learning of a student controller that only uses sensors available on the real robot. Similarly in [25], the student-teacher paradigm is used to learn in-hand reorientation. The teacher used the object orientation to quickly learn appropriate actions, after which the student was trained to match the teacher’s outputs while relying on occluded visual observations of the objects. Instrumentation like in [20]–[23] can be a way to obtain similar privileged information directly in the real world, potentially allowing for faster learning and/or more performant policies and avoiding the need for sim-to-real transfer, which already is a significant challenge on its own.

This work presents foundational results uncovering the potential of instrumentation for IL through a case study focusing on the task of clothes hanger insertion. We show that (1) a black-box IL policy can recognise the importance of instrumentation signals without explicit guidance, (2) including instrumentation in the training data for an IL policy can increase both the policy’s task awareness as well as overall performance, and (3) that rollouts from such an “expert” policy can be used as extra training data to enhance the performance of a “student” policy that cannot rely on instrumentation.

II. MATERIALS & METHODS

A. Task Description

Cloth manipulation is a challenging research topic in robotics requiring advanced methods for modelling, perception, and control [26]. The field comprises a variety of subtasks, like collection, unfolding, folding, hanging, etc. Whereas many systems have been proposed for, e.g., unfolding and folding, the task of clothes hanger insertion has received limited attention [27], [28]. We choose clothes hanger insertion as a testbed for instrumentation, because the (mostly) rigid clothes hanger lends itself to sensor integration, while the task as a whole is challenging and rarely addressed in the state-of-the-art of cloth manipulation.

Fig. 1 shows the progression of the insertion task. In order to alleviate data requirements and not lose focus from the core research questions of this work regarding the

potential of instrumentation, some constraints apply to task initialisation:

- Both the T-shirt and the clothes hanger are already held by the robot arms.
- We only use one T-shirt and one clothes hanger, always held in the same location.

Once started, the robot inserts the first leg of the hanger in the open collar of the T-shirt, and lifts the clothes hanger until it is in a good position to perform the insertion of the second leg. After the second insertion, the T-shirt is released so that it hangs only from the clothes hanger. For evaluation rollouts, “task success” is defined as the autonomous execution of these stages, concluding with the T-shirt hanging only from the clothes hanger, on both shoulders.

B. Hardware setup

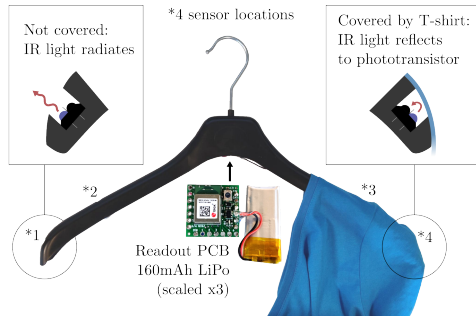
1) *Instrumented Clothes Hanger*: A standard clothes hanger is instrumented by integration of four TCRT5000 reflective infrared (IR) range sensors, see Fig. 2. The TCRT5000 contains an IR light-emitting diode (LED), and an IR phototransistor (PT). When the clothes hanger is not covered, the IR light radiates away, but if the T-shirt covers it, the light reflects back to the PT, causing a large change in the photocurrent, as can be seen in the top curves of Fig. 1. For readout, we use a printed circuit board (PCB) from our open-source, modular, wireless Smart Textile [23], featuring an nRF52840 microcontroller. The PCB, as well as a 160mAh LiPo battery, easily fit inside the hollow clothes hanger. Data is communicated to a workstation over Bluetooth Low Energy (BLE).

2) *Robot Setup*: The robot system is shown in Fig. 3a. It features two UR5e collaborative robot arms: the left holds the T-shirt, the right moves the clothes hanger. A Schunk EGK-40-MB-M-B gripper is mounted to each arm. Standard Robotiq 2F-85 fingertips are fitted to the Schunk grippers by means of 3D-printed mechanical adapters. The left robot arm has a RealSense D405 camera attached to its wrist to get a close-up view of the collar opening of the T-shirt. An externally mounted Zed 2i camera provides a view of the entire scene.

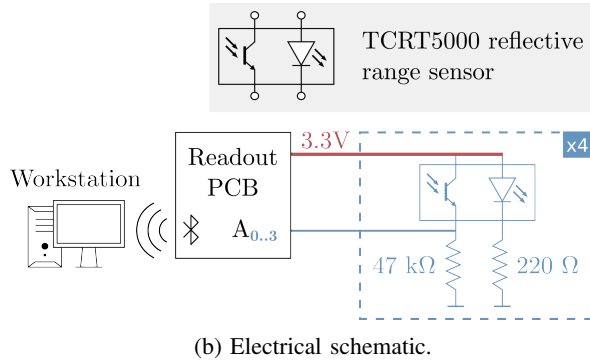
3) *Teleoperation Setup*: We use a Gello [29] arm (Fig. 3b) to control the joint positions of the right UR5e, the left UR5e is static. The Gello arm trigger is configured to control the opening of the Schunk gripper on the left UR5e, the opening of the Schunk gripper on the right UR5e is fixed.

C. Policy Architecture

For the policies trained in this work, we use a Diffusion Policy [30] architecture with a ResNet18 vision backbone and a CNN-based noise prediction network. We set the hyperparameters to the default values from [30] unless specified otherwise. The model predicts the joint positions of the right UR5e and the gripper opening of the left UR5e, given an observation that consists of 720x500x3 images from both wrist and scene camera, and a state vector. The state vector is composed of the six joint poses of the right UR5e, the gripper opening of the left UR5e, and optionally the four sensor



(a) Sensor integration and working principle. The electronics fit inside the hollow clothes hanger.



(b) Electrical schematic.

Fig. 2: Instrumented clothes hanger with integrated infrared (IR) reflective range sensors. The range sensors give a strongly different signal when the clothes hanger is covered, compared to when it is not.

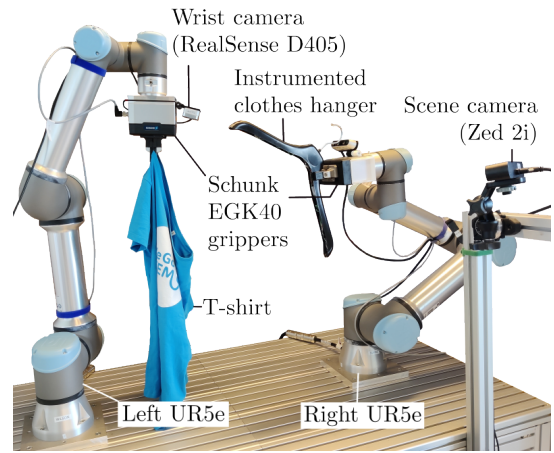
readings of the clothes hanger: in the following, policies with and without access to instrumentation data are compared. The latter may be denoted as “vision-only” policies, even though the robot joint poses and gripper opening are still included in the state. The model predicts a chunk of 16 actions, of which 8 are executed before predicting another chunk. At a control frequency of 10 Hz, these 8 actions correspond to 0.8 s of wall-clock time. We train all models for 100 000 steps with a batch size of 32, which takes about 11 h on an NVIDIA RTX4090 GPU, inference takes 270 ms.

In this work, a total of seven policies are trained. Policies for which the clothes hanger sensor data is included in the state vector are denoted as Π_{instr}^t , vision-only policies as Π_{vis}^t , with t being the number of demonstrations that the policy was trained on.

D. Data Collection

Three different types of datasets are collected in this work, each following their own protocol: (1) human demonstrations are collected through teleoperation (subsection II-D.1); (2) policy rollouts are collected to estimate success rates (subsection II-D.2); (3) policy rollouts from a policy relying on instrumentation are added to the teleoperation data, forming an enhanced, larger dataset to train better vision-only policies (subsection II-D.3). A single episode (demo or rollout) can take four forms:

Type I: Initialisation exactly as depicted in Fig. 1 and



(a) Robot setup.



(b) The Gello [29] teleoperation arm.

Fig. 3: Experimental setup for learning clothes hanger insertion.

3a, with the right UR5e starting from a fixed home pose. The task is executed to completion.

Type II: The right arm starts from a “missed first insertion”, i.e., the first leg of the clothes hanger is near to, but below the level of, the collar of the T-shirt. The task is executed to completion.

Type III: The right arm starts from a “missed first insertion”, the episode ends right after the first insertion.

Type IV: The right arm starts from its home pose, the episode ends right after the first insertion.

While Type I represents the full task, Types II, III and IV are focused on achieving the first insertion, as this stage of the task was found to be much more difficult than the second insertion, see the failure mode analysis in subsection III-B. The initial state of the T-shirt is randomised by manually lifting the shoulder that is not grasped by the robot and letting it go. In section III, Fig. 4, we will show that this procedure provided sufficient randomisation. Datasets are available on Zenodo [10.5281/zenodo.17122216].

1) *Teleop training data:* In total, 180 human demonstrations of the task shown in Fig. 1 are collected. Three subsets of this data are defined:

#train=180: This is the full teleop set, comprising 50 Type I demonstrations, 120 Type II demonstrations, 10 Type III demonstrations.

#train=100: 20 Type I demonstrations, 80 Type II demonstrations.

#train=50: 50 Type II demonstrations.

The **#train=180** set comprises 47 040 observations, which accounts for 1.31 h of data collection. However, this excludes resetting the environment between demonstrations, compressing and saving each demonstration, and occasionally reinitialising the entire data collection pipeline after e.g. a collision. Overall, we estimate a data collection time of 3.5 h.

2) *Policy rollouts for evaluation metrics:* All policy rollouts used to calculate success rates are of Type I. They are given at least one minute to complete an insertion, both for the first and second. If after one minute the policy is not reaching new states (anecdotally, this mostly occurs when the clothes hanger is not touching the T-shirt), the rollout is ended and declared a failure. If the policy takes longer than two minutes for an insertion, the rollout is ended no matter how close it is to completion.

3) *Policy rollouts for enhanced dataset:* In the Results section (III), it will be shown that including instrumentation data increases both a policy’s task awareness and overall performance. However, policies relying on instrumentation cannot be deployed directly in the field since we do not envision every household object to have embedded sensors: we need ways to omit the instrumentation data without losing performance. In this work, we use rollouts from a policy relying on instrumentation as extra training data to obtain a more performant vision-only policy.

Specifically, we add the successful evaluation rollouts (Type I) of Π_{instr}^{180} to the **#train=180** set. In addition, we review the failures of Π_{vis}^{180} and recreate their initialisations as closely as possible by overlaying their initial camera observations with a live camera feed. Then, we let Π_{instr}^{180} attempt Type IV rollouts on each of the difficult initialisations, until five successful demonstrations are obtained, with a maximum of ten attempts per initialisation. As such, we obtain an enhanced dataset, **#train=237**, for training a better vision-only policy $\Pi_{\text{vis}+}^{237}$ (the extra “+” subscript emphasises the enhanced training data compared to other Π_{vis} policies). Note that we do not filter the Π_{instr}^{180} rollouts for optimal behaviour: mistakes and corrective behaviours are included as long as the rollout is successful. See subsection III-A for more details on the additional 57 trials.

III. RESULTS

Seven policies are trained: Π_{instr}^{50} and Π_{vis}^{50} on the **#train=50** set, Π_{instr}^{100} and Π_{vis}^{100} on the **#train=100** set, Π_{instr}^{180} and Π_{vis}^{180} on the **#train=180** set, and $\Pi_{\text{vis}+}^{237}$ on the enhanced **#train=237** set. Π^{50} and Π^{100} policies are evaluated with 20 rollouts each, Π^{180} policies with 30, and the $\Pi_{\text{vis}+}^{237}$ policy with 40 rollouts. The initial states of the evaluation sets are visualised in Fig. 4. This figure overlays the collar of the T-shirt as seen from the wrist camera at the start of each rollout for each of the evaluation sets. It indicates that the distribution of initial states is sufficiently random, and that failures and successes are not biased towards any type of initial state.

A. Policy success rates

The experimental end-to-end success rates are shown in Fig. 5. For a given training set size, a policy with access

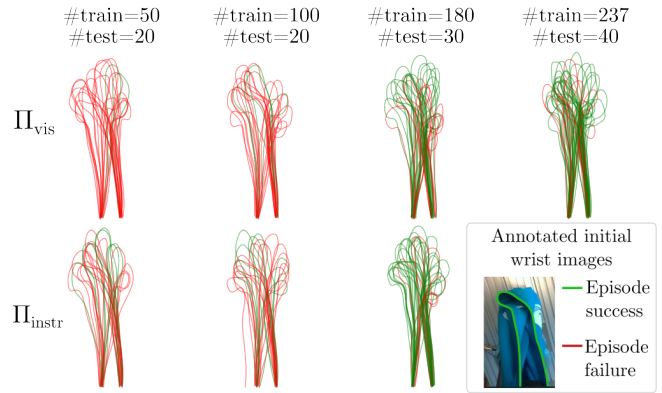


Fig. 4: Visualisation of each test set. The curves are traces of the collar of the T-shirt as seen from the wrist camera at the start of an episode.

to instrumentation data consistently outperforms a vision-only policy by 14 to 25 %pt. All 23 successful Π_{instr}^{180} rollouts from the test set, as well as 34 additional Π_{instr}^{180} rollouts targeting the 11 Π_{vis}^{180} failures according to the formalism described in subsection II-D.3, are added to the **#train=180** set to obtain the **#train=237** set. The vision-only policy $\Pi_{\text{vis}+}^{237}$ trained on **#train=237** outperforms Π_{vis}^{180} by 12 %pt. To better frame these results, we perform a Bayesian analysis [31]. Task success is treated as a Bernoulli distribution with unknown parameter p , the probability of success, which we can estimate given the experimental results. According to Bayes’ theorem, the probability density function (PDF) of p given s measured successes out of N rollouts, $P(p|s)$, can be expressed as:

$$P(p|s) = P(s|p) \frac{P(p)}{P(s)}. \quad (1)$$

Assuming a uniform distribution on p , $P(p)$ is equal to 1. $P(s|p)$, the PDF of the number of successes when running N Bernoulli experiments given success rate p , follows a binomial distribution:

$$P(s|p) = p^s (1-p)^{N-s} \binom{N}{s}. \quad (2)$$

Lastly, $P(s)$ is calculated as:

$$\begin{aligned} P(s) &= \int_0^1 P(s,p) dp = \int_0^1 P(s|p) P(p) dp \\ &= \int_0^1 P(s|p) dp \stackrel{(2)}{=} \binom{N}{s} \int_0^1 p^s (1-p)^{N-s} dp \\ &= \binom{N}{s} B(s+1, N-s+1) \\ &= \binom{N}{s} \frac{\Gamma(s+1) \Gamma(N-s+1)}{\Gamma(N+2)} \\ &= \frac{N!}{s!(N-s)!} \frac{s!(N-s)!}{(N+1)!} = \frac{1}{N+1}. \end{aligned} \quad (3)$$

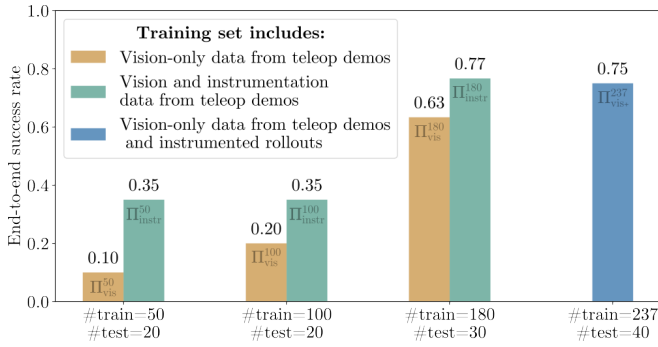


Fig. 5: End-to-end success rates for diffusion policies trained on different dataset sizes, including different modalities. Instrumented policies outperform their vision-only counterparts. A vision-only policy can be improved by adding rollouts from an instrumented policy to the training data.

with B and Γ the well-known Beta and Gamma functions, respectively. We can now write equation 1 as:

$$P(p|s) = p^s (1-p)^{N-s} \frac{(N+1)!}{s!(N-s)!}. \quad (4)$$

It follows that $P(p|s) \sim \text{Beta}(s+1, N-s+1)$, or, with \hat{p} the experimentally determined success rate as reported in Fig. 5: $P(p|\hat{p}) \sim \text{Beta}(N\hat{p}+1, N(1-\hat{p})+1)$. This way, we can estimate the distribution of p for each Π . Using Monte Carlo simulation, we obtain:

$$\begin{cases} P[p_{\text{instr}}^{50} \sim \text{Beta}(8, 14) > p_{\text{vis}}^{50} \sim \text{Beta}(3, 19)] = 96.7\% \\ P[p_{\text{instr}}^{100} \sim \text{Beta}(8, 14) > p_{\text{vis}}^{100} \sim \text{Beta}(5, 17)] = 84.7\% \\ P[p_{\text{instr}}^{180} \sim \text{Beta}(24, 8) > p_{\text{vis}}^{180} \sim \text{Beta}(20, 12)] = 86.5\% \\ P[p_{\text{instr}}^{237} \sim \text{Beta}(31, 11) > p_{\text{vis}}^{180} \sim \text{Beta}(20, 12)] = 85.3\%. \end{cases} \quad (5)$$

B. Failure Modes

We distinguish the following failure modes for unsuccessful policy rollouts:

Collision[1st insertion]: A collision occurs between clothes hanger, gripper, and/or wrist camera before a successful first insertion.

Drop[1st insertion]: The gripper of the left UR5e opens and drops the T-shirt before a successful first insertion has been performed (see Fig. 6a).

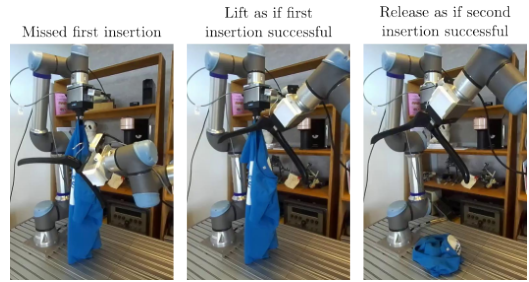
Stuck[1st insertion]: The policy attempts the first insertion, but misses the collar of the T-shirt and gets stuck.

Pulled drop: The policy attempts the first insertion, but pulls the T-shirt down and out from between the fingertips of the left UR5e.

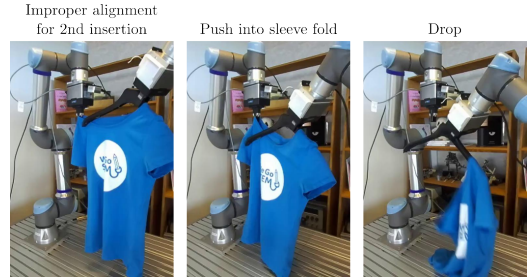
Failed lift: The policy succeeds in the first insertion, but does not move on to lift the T-shirt.

Stuck[2nd insertion]: The policy gets stuck right before the second insertion.

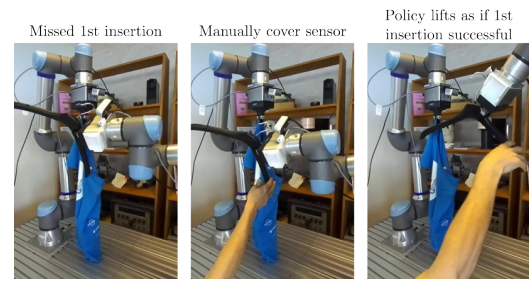
Collision[2nd insertion]: A collision occurs between clothes hanger, gripper, and/or wrist camera during an attempt of the second insertion.



(a) A vision-only policy often executes robot movements without concern for the state of the T-shirt, resulting in a dropped T-shirt.



(b) To an instrumented policy, a sleeve draped over the clothes hanger can look similar to a properly inserted hanger.



(c) An instrumented policy relies more strongly on the sensor data than the visual feed. After a failed first insertion, the policy realises it cannot continue. When manually covering the sensor, the policy attempts to lift, disregarding visual cues.

Fig. 6: Policy errors and behaviours related to task awareness.

Drop[2nd insertion]: The gripper of the left UR5e opens and drops the T-shirt after an unsuccessful attempt at the second insertion (see Fig. 6b).

Table I compiles the incidence of each failure mode for each of the seven policies. The colour scale is normalised with respect to the number of failures in the test set for each policy.

IV. DISCUSSION

The experimental success rates in Fig. 5 show that the instrumented policies consistently outperform their vision-only counterparts. The gap is largest (25 %pt) for the smallest train set ($\#train=50$). For larger train sets, the gap is 14-15 %pt. Using Bayesian analysis (see equation 5), which effectively incorporates both the experimental success rate and the sample size into one figure of merit, we can make the following statement. According to our data, it is at least 84.7% certain that an instrumented policy outperforms a

TABLE I: Policy failure mode incidence (color scale normalised per row).

Policy	# Rollouts in test set	# Failures in test set	Collision [1st insertion]	Drop [1st insertion]	Stuck [1st insertion]	Pulled drop	Failed lift	Stuck [2nd insertion]	Collision [2nd insertion]	Drop [2nd insertion]
Π_{vis}^{50}	20	18	2	3	11	0.0	0.0	0.0	0.0	2
Π_{instr}^{50}	20	13	0.0	0.0	9	1	0.0	0.0	0.0	3
Π_{vis}^{100}	20	16	0.0	7	6	0.0	0.0	1	2	0.0
Π_{instr}^{100}	20	13	0.0	0.0	6	0.0	0.0	6	1	0.0
Π_{vis}^{180}	30	11	0.0	0.0	11	0.0	0.0	0.0	0.0	0.0
Π_{instr}^{180}	30	7	0.0	0.0	3	3	0.0	1	0.0	0.0
Π_{vis+}^{237}	40	10	0.0	0.0	7	1	1	0.0	1	0.0

vision-only policy. However, an instrumented policy cannot easily be deployed in the field: we do not envision every clothes hanger in the bedroom closet having embedded sensors. In this work, we used the instrumented policy Π_{instr}^{180} to gather additional data, part of which was specifically targeted to failures of the vision-only policy Π_{vis}^{180} . A new vision-only policy Π_{vis+}^{237} trained on this enhanced data set showed a performance similar to Π_{instr}^{180} —it makes sense that the student does not outperform the teacher—and outperformed Π_{vis}^{180} by 12 %pt. In the Bayesian interpretation (equation 5), we are 85.3 % certain that Π_{vis+}^{237} outperforms Π_{vis}^{180} .

Apart from success rates, we also note substantial differences in the policy failure modes and behaviours. For Π_{vis}^{50} , 5 out of 18 failures occurred when the left gripper dropped the T-shirt at a point where the hanger insertion was not completed (**Drop [1st insertion] + Drop [2nd insertion]**). For Π_{vis}^{100} 7 out of 16 failures are **Drop [1st insertion]**: the policy seems to execute robot movements with little regard for the state of the T-shirt (see Fig. 6a). This indicates little task awareness of the vision-only policy at small training set sizes. Π_{instr}^{50} had three **Drop [2nd insertion]** failures out of 13, while no more **Drop** failures occurred for Π_{instr}^{100} . This tendency of getting stuck rather than failing catastrophically opens the door for more effective human takeovers during interactive IL like [14]–[16].

Importantly, the **Drop** failures of Π_{instr}^{50} were caused by the outer sensor of the second hanger leg (*1 in Fig. 2a) being covered by a folded sleeve of the T-shirt (see Fig. 6b). This shows how strongly the Π_{instr} policies rely on the instrumentation data. Additional evidence is shown in Fig. 6c: Π_{instr}^{50} misses the first insertion and realises it is not ready to lift. When we manually cover the sensor that should be in the T-shirt at this point, the visual cues are ignored and the policy proceeds to lift. This is an important and non-trivial result underpinning the potential of instrumentation: a black-box IL policy can recognise the importance of instrumentation signals without explicit guidance.

These results favour instrumentation for IL, but we must keep in mind the question of whether the effort spent implementing instrumentation had not better been redirected to collecting additional teleop demonstrations [12]. In this case study, collecting an additional 57 teleop demonstrations would take less than 1.5h, which is less than the development time of the instrumentation system. However,

while the integration of infrared range sensors is specifically designed for the task of clothes hanger insertion, the modular readout PCB has already proven to be applicable to multiple tasks [23], [32]. Hence, expanding this open-source platform of instrumentation hardware can accelerate development of instrumented objects and enable instrumented learning.

Even still, instrumentation requires a certain data scale to be beneficial. Building intuition towards where this turning point lies is a key point of interest for future research. However, before we scale up robot learning experiments to larger datasets, it pays to investigate alternative methods of leveraging instrumentation data during training, while obtaining a policy that can be deployed without instrumentation input. Several methods are proposed in section V.

Furthermore, we believe that the performance gap between instrumented and vision-only policies may increase with less constraints on the task, i.e. more types of clothes hanger, more T-shirts, more variation in the position of the T-shirts. These variations highly impact the vision modality, while the instrumentation modality is relatively unbothered: even with varying IR absorption between different coloured T-shirts, the difference in sensor readings between covered and uncovered states is tremendous.

V. CONCLUSION & FUTURE WORK

This work presents foundational results uncovering the potential of instrumentation for IL through a case study focusing on clothes hanger insertion. We developed an instrumented clothes hanger by integrating IR reflective range sensors, which provide a strong signal when the clothes hanger is covered compared to when it is not. This way, the instrumentation provides a clear indication of task progression, which cannot be directly inferred from other modalities such as vision or, potentially, tactile sensing. A dataset of 180 teleop demonstrations of the hanger insertion task was collected. Three different subsets of this dataset were defined, and for each subset two Diffusion Policy models were trained: one including instrumentation data in the state vector, a second excluding instrumentation data. First and foremost, we found that instrumented policies recognise the importance of instrumentation signals without explicit guidance, even prioritising them above visual cues. Second, instrumented policies outperform their vision-only counterparts by 14-25 %pt. Lastly, we showed that adding

rollouts from an “expert” instrumented policy to the teleop dataset and training a “student” vision-only policy on this enhanced dataset increases the performance of the vision-only student to the level of the instrumented expert.

A critical note is that in this case study the performance gap between instrumented and vision-only policies is not large enough to warrant the effort spent on implementing instrumentation. However, the readout PCB used in this work has already proven valuable for multiple tasks, so further development of this open-source instrumentation platform can enable efficient instrumented learning. Furthermore, we expect the performance gap to increase when allowing for more variation in the task. Finally, using an instrumented policy to obtain expert rollouts is just one possible way to leverage instrumentation to achieve a more performant vision-only policy. We outline several approaches to be developed in future work:

- **Soft sensor:** An additional, specialised model is trained to predict the sensor values from the camera images. At inference time, the predictions of this “soft sensor” [33] are used for the instrumentation inputs of the action policy. This approach effectively decomposes the task in (1) explicit state estimation and (2) action selection given a certain state.
- **Pretraining:** First train a ResNet18 model to predict the sensor values given the camera images, and then use it as the vision backbone for the Diffusion Policy network. By pretraining on the sensor data, we aim to capture essential information about the goal and task progression in the backbone network, which provides the action policy with a head-start for training. Previous work has shown that pretraining a visuo-tactile encoder and using it as the vision backbone for a vision-only IL policy can drastically increase performance [34].
- **Additional training loss:** We train the network as a whole to predict both robot actions and the sensor data, either by using a concatenation of the sensor values and teleop actions as training labels, or by adding a second prediction head and combining its loss with that of the robot action prediction head. In essence, this forces the model to explicitly predict whether or not the clothes hanger is covered, which is a crucial indicator of task progression.

Once we have a broad understanding of how instrumentation data can be leveraged to speed up robot learning, we can conduct experiments at larger data scales and discover when instrumented learning can reduce overall human effort in obtaining functional robot manipulation policies.

VI. ACKNOWLEDGMENT

GPT-5 was used to write the first draft of the abstract. GPT-5 and Writefull were used sparingly throughout the manuscript to improve specific sentences and wordings.

REFERENCES

[1] A. Billard, *et al.*, “A roadmap for ai in robotics,” *Nature Machine Intelligence*, vol. 7, no. 6, p. 818–824, June 2025. [Online]. Available: <http://dx.doi.org/10.1038/s42256-025-01050-6>

[2] K. Goldberg, “Good old-fashioned engineering can close the 100,000-year “data gap” in robotics,” *Science Robotics*, vol. 10, no. 105, p. eaea7390, 2025. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.aea7390>

[3] J. Barreiros, *et al.*, “A careful examination of large behavior models for multitask dexterous manipulation,” *arXiv preprint arXiv:2507.05331*, 2025.

[4] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, “A survey of robot learning from demonstration,” *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 469–483, 2009. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0921889008001772>

[5] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, “Learning Fine-Grained Bimanual Manipulation with Low-Cost Hardware,” in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[6] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, “Diffusion policy: Visuomotor policy learning via action diffusion,” *The International Journal of Robotics Research*, 2024.

[7] K. Black, *et al.*, “ π_0 : A vision-language-action flow model for general robot control,” 2024. [Online]. Available: <https://arxiv.org/abs/2410.24164>

[8] M. J. Kim, *et al.*, “Openvla: An open-source vision-language-action model,” *2024 Conference on Robot Learning*, vol. abs/2406.09246, 2024. [Online]. Available: <https://api.semanticscholar.org/CorpusID:270440391>

[9] E. Collaboration, *et al.*, “Open x-embodiment: Robotic learning datasets and rt-x models,” 2024. [Online]. Available: <https://arxiv.org/abs/2310.08864>

[10] A. Khazatsky, *et al.*, “Droid: A large-scale in-the-wild robot manipulation dataset,” in *Proceedings of Robotics: Science and Systems*, 2024. [Online]. Available: <https://arxiv.org/abs/2403.12945>

[11] H. Walke, *et al.*, “Bridgedata v2: A dataset for robot learning at scale,” 2024. [Online]. Available: <https://arxiv.org/abs/2308.12952>

[12] S. Mirchandani, S. Belkhal, J. Hejna, E. Choi, M. S. Islam, and D. Sadigh, “So you think you can scale up autonomous robot data collection?” 2024. [Online]. Available: <https://arxiv.org/abs/2411.01813>

[13] C. Celemin, R. Pérez-Dattari, E. Chisari, G. Franzese, L. de Souza Rosa, R. Prakash, Z. Ajanović, M. Ferraz, A. Valada, and J. Kober, “Interactive imitation learning in robotics: A survey,” 2022. [Online]. Available: <https://arxiv.org/abs/2211.00600>

[14] N. Ingelhart, J. Munkeby, M. C. Welle, M. Moletta, and D. Kragic, “Real-time operator takeover for visuomotor diffusion policy training,” 2025. [Online]. Available: <https://arxiv.org/abs/2502.02308>

[15] M. Kelly, C. Sidrane, K. Driggs-Campbell, and M. J. Kochenderfer, “Hg-dagger: Interactive imitation learning with human experts,” 2019. [Online]. Available: <https://arxiv.org/abs/1810.02890>

[16] Y. Dai, J. Lee, N. Fazeli, and J. Chai, “Racer: Rich language-guided failure recovery policies for imitation learning,” 2024. [Online]. Available: <https://arxiv.org/abs/2409.14674>

[17] M. Sakr, Z. J. Li, H. F. M. Van der Loos, D. Kulić, and E. A. Croft, “Quantifying demonstration quality for robot learning and generalization,” *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9659–9666, 2022.

[18] M. Bilal, N. Lipovetzky, D. Oetomo, and W. Johal, “Beyond success: Quantifying demonstration quality in learning from demonstration,” in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 5120–5127.

[19] C. Xu, Q. Li, J. Luo, and S. Levine, “Rldg: Robotic generalist policy distillation via reinforcement learning,” 2024.

[20] K. Junge, C. Pires, and J. Hughes, “Lab2field transfer of a robotic raspberry harvester enabled by a soft sensorized physical twin,” *Communications Engineering*, vol. 2, no. 1, p. 40, Jun 2023.

[21] OpenAI, *et al.*, “Solving rubik’s cube with a robot hand,” 2019.

[22] A. Verleysen, T. Holvoet, R. Proesmans, C. Den Haese, and F. wyffels, “Simpler learning of robotic manipulation of clothing by utilizing diy smart textile technology,” *Applied Sciences*, vol. 10, no. 12, 2020. [Online]. Available: <https://www.mdpi.com/2076-3417/10/12/4088>

[23] R. Proesmans, A. Verleysen, R. Vleugels, P. Veske, V.-L. De Gussemme, and F. wyffels, “Modular piezoresistive smart textile for state estimation of cloths,” *Sensors*, vol. 22, no. 1, 2022.

[24] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, “Learning quadrupedal locomotion over challenging terrain,” *Science Robotics*, vol. 5, no. 47, Oct. 2020. [Online]. Available: <http://dx.doi.org/10.1126/scirobotics.abc5986>

- [25] T. Chen, M. Tippur, S. Wu, V. Kumar, E. Adelson, and P. Agrawal, "Visual dexterity: In-hand reorientation of novel and complex object shapes," *Science Robotics*, vol. 8, no. 84, p. eadc9244, 2023. [Online]. Available: <https://www.science.org/doi/abs/10.1126/scirobotics.adc9244>
- [26] A. Longhini, *et al.*, "Unfolding the literature: A review of robotic cloth manipulation," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 8, 2025.
- [27] Y. Chen, S. Wei, B. Xiao, J. Lyu, J. Chen, F. Zhu, and H. Wang, "Robohanger: Learning generalizable robotic hanger insertion for diverse garments," *IEEE Robotics and Automation Letters*, vol. 10, no. 9, pp. 8922–8929, 2025.
- [28] Y. Koishihara, S. Arnold, K. Yamazaki, and T. Matsubara, "Hanging work of t-shirt in consideration of deformability and stretchability," in *2017 IEEE International Conference on Information and Automation (ICIA)*, 2017, pp. 130–135.
- [29] P. Wu, Y. Shentu, Z. Yi, X. Lin, and P. Abbeel, "Gello: A general, low-cost, and intuitive teleoperation framework for robot manipulators," 2024.
- [30] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," *The International Journal of Robotics Research*, 2024.
- [31] H. Kress-Gazit, K. Hashimoto, N. Kuppuswamy, P. Shah, P. Horgan, G. Richardson, S. Feng, and B. Burchfiel, "Robot learning as an empirical science: Best practices for policy evaluation," 2024. [Online]. Available: <https://arxiv.org/abs/2409.09491>
- [32] R. Proesmans, T. Lips, and F. Wyffels, "Instrumentation for better demonstrations: A case study," 2025. [Online]. Available: <https://arxiv.org/abs/2504.18481>
- [33] Y. S. Perera, D. Ratnaweera, C. H. Dasanayaka, and C. Abeykoon, "The role of artificial intelligence-driven soft sensors in advanced sustainable process industries: A critical review," *Engineering Applications of Artificial Intelligence*, vol. 121, p. 105988, 2023. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0952197623001720>
- [34] A. George, S. Gano, P. Katragadda, and A. B. Farimani, "Vital pretraining: Visuo-tactile pretraining for tactile and non-tactile manipulation policies," 2024. [Online]. Available: <https://arxiv.org/abs/2403.11898>