2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)
June 28-30, 2023. Seattle, Washington, USA

2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) | 978-1-6654-7633-1/23/$31.00 ©2023 IEEE | DOI: 10.1109/AIM46323.2023.10196256

Development of an Autonomous, Explainable, Robust Robotic System for Electric Vehicle Battery Disassembly

Yisheng Zhang[1], Hengwei Zhang[1], Zhigang Wang[2], Shengmin Zhang[1], Huaicheng Li[1], Ming Chen[1]

Abstract— **The vigorous growth of the electric vehicle industry calls for efficient disassembly of used electric vehicle batteries (EVBs). Screw disassembly by robots remains a challenge due to the uncertainties in this task. In this paper, we designed an architecture of NeuroSymbolic task and motion planning, which uses neural predicates to map the sensor into a quasi-symbolic state and schedules action primitives autonomously based on current state and goal state. This architecture guarantees autonomy and explainability which is important in human-robot hybrid disassembly pipeline. In primitive implementation, a customized end-effector, accurate vision-based and force-based pose estimation are enabled to ensure the robustness of the system. The experiment shows that the proposed system can achieve 100% success rate in lab environment. We will deploy and evaluate it in the real factory environment in the future.**

I. INTRODUCTION

With the rapid development of the electric vehicle industry, improving the efficiency of electric vehicle batteries (EVBs) disassembly is imperative. However, due to the numerous variants in the design, brand, integrity, and end-of-life (EOL) of used batteries, the disassembly process is far more complex than the assembly process. According to a recent survey [1], automatic disassembly of the whole process has not been feasible. People have begun to use a human-robot hybrid mode to improve disassembly efficiency, that is, simple, repetitive tasks such as screw disassembly(accounting for about 40% of the entire battery disassembly process [2], [3]) are performed by robots, and tasks requiring high flexibility, such as removing cables are undertaken by humans. But disassembling the screws by the robot in used EVBs is still challenging. Unlike robots in the automatic assembly pipeline, robots in the disassembly pipeline cannot complete the task by pre-programmed actions. The control process needs to be autonomous, explainable, and robust. Autonomy enables the robot to choose the proper action sequence according to the actual situation, ensuring high efficiency while adapting to uncertain environments. And robustness ensures a high success rate of disassembling operations. Explainability informs people of what the robot is doing and why, so they can cooperate with it or provide assistance when necessary.

However, the existing researches fail to meet the above fundamental requirements well. Some customized systems [4], [5] have been developed. For lack of autonomy, they are only suitable for static environments (clean, non-deformed, and specific types of batteries) and thus not popularized. Some follow-up studies attempted to expand the application range of the system by ingeniously designing the hardware platform and introducing more sensors or ingenious end effector designs [6]–[11]. But their success rates of operations are still not satisfactory.

Currently, the new developments utilize the immense progress in machine learning to encapsulate uncertainty models and support further advances in adaptive and robust control [12]. Some researchers train deep neural networks (DNN) to determine objects' types and positions and then decide the following action plan based on them [13]–[16]. Since obtaining high-quality labeled data in EVBs disassembly is not easy, the stability and accuracy of positioning often fail to meet the requirements. Some researchers try to learn a control policy of robots through Reinforcement learning(RL) [17], [18]. These methods only work well when they can transfer from simulator to reality, which is challenging to realize. Moreover, people can't cooperate with this kind of robot because they don't know what it is doing or why.

Therefore, we proposed an architecture of NeuroSymbolic task and motion planning (NeuroSymbolic TAMP) to enable an Autonomous, Explainable, Robust Robotic System for EVBs Disassembly, and to tackle the uncertainty issues in the disassembly process under the unstructured conditions, with the integration of the disassembly tasks decision and the robot motion perception and control. We expect the system to be part of the human-robot hybrid disassembly pipeline and replace workers in tasks such as screw disassembly. NeuroSymbolic TAMP uses Planning Domain Define Language (PDDL) [19]–[21] to define disassembly primitives and get the entire procedure through on-site reasoning. In particular, we introduce the neural predicates, which use neural networks to map the sensor data to quasi-symbolic states. The planner engine can conclude the action plan based on neural predicates. In our previous paper [22], we verified that the NeuroSymbolic TAMP could adapt to the dynamic environment autonomously. This paper focuses on improving the system's usability in real scenarios. We enhance the implementation of primitives and extend the primitives in the system. We leverage the capability of the robot to combine deep learning with canonical control theory to realize high-accurate primitives. In this paper, more primitives are introduced to utilize force/torque sensor data. These extensions ensure that the robot can complete the disassembly task with a high success rate via multimodal

This work was supported by the Ministry of Industry and Information Technology of China for financing this research within the program ”2021 High Quality Development Project (TC210H02C)”

The authors are with [1] Shanghai Jiaotong University, School of Mechanical Engineering, China. [2]Intel Labs China, China. Corresponding author: Ming Chen, email: mingchen@sjtu.edu.cn

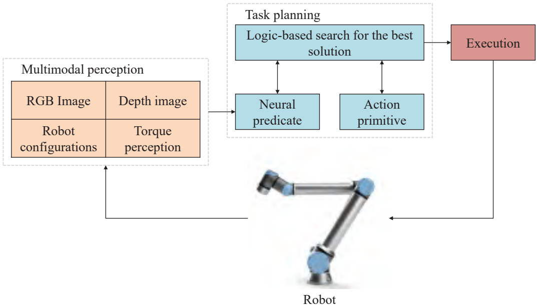


Fig. 1. Intelligent disassembly system based on NeuroSymbolic

perception in dynamic, unstructured environments. On-site experiments show that the system can work robustly for EVBs disassembly tasks.

II. PDDL-based NeuroSymbolic TAMP

The architecture of the NeuroSymbolic TAMP is as Figure 1 shows. The task planning module is a symbolic logic system in frame, of which the ”grounding” is realized by neural predicates derived from neural networks taking multimodal signals as input. We design and implement 8 action primitives by analyzing workers’ skills. With neural predicates, each primitive is formally defined in PDDL as a composition of executable preconditions and execution results (see https://sites.google.com/view/evb-disassembly). When performing task planning, the system automatically uses neural predicates to convert sensor data into quasi-symbolic state descriptions. Benefiting from this encapsulation, the system acquires the optimal sequence of primitives achieving the goal state from the current state using logical search algorithms [22] and then sequentially executes these primitives to complete the task. Unlike classic symbolic planning, the system continuously perceives the environment and updates its state during execution, ensuring that it can re-plan immediately once the state is inconsistent with expectations.

NeuroSymbolic TAMP guarantees explainability at the system level, since planning and execution within the scope of PDDL makes each decision and execution of the robot understandable. The system performs task planning based on real-time state awareness, and can self-correct relying on neural predicates instead of artificially abstracted and unchangeable states in advance, which demonstrates its autonomy. The system also integrates some prior knowledge, such as the approximate distribution of screws generalized from representative battery pack models. This paper focuses only on the implementation of single screw disassembly. In such an architecture, the primitives are the crux to ensure the system’s robustness. In the next section, we will discuss the implementation of primitives in detail.

III. Implementation and Challenges of Primitives

In reality, screws can’t be detected and located accurately owing to the irregular geometry, inconsistent texture, and distorted depth information. We have ensured that the system can continuously and stably complete the screw disassembly task at the three levels of end effector design, vision-based pose estimation, and force perception.

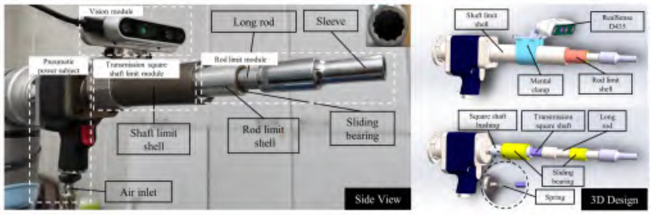


Fig. 2. (Left) The structure of real machine. (Right) Internal structure of the nut runner.

A. Customized end effector

According to the characteristics of the screw disassembly task, we designed a pneumatic torque actuator with compliant mechanism and vision sensing (as shown in Figure 2), including a pneumatic power subject, a square shaft transmission limit module, a rod limit module and a vision module. The structure of the pneumatic power subject is similar to that of a pneumatic impact wrench with an air inlet. The transmission square shaft limit module consists of a shaft limit shell, a transmission square shaft, a sliding bearing, a spring and a square shaft bushing. This compliant mechanism with a spring provides freedom of axial movement for the long rod underneath while transmitting the torque output by the pneumatic power subject to it. This design not only prevents rigid collision when loosening the screw, but also compensates for the depth positioning error during insertion. The long rod makes the actuator suitable for the complex disassembly environment inside the battery pack, and replaceable sleeves provide possibility to remove more types of screws. The main component of the vision module is a fixed Intel RealSense D435 camera. Its view clearly shows the relative position of the screw and the sleeve, which is conducive to accurate positioning and judgement of neural predicates.

In addition, for the hexagonal screws discussed in this paper, a 12-point sleeve is applied to achieve successful clamping only with a slight rotation. In summary, the customized end effector provides good passive compliance for insertion with a simple structure.

B. Vision-based Pose Estimation

In this paper, we use an object detection algorithm based on YOLOv5 and a pose estimation algorithm based on RANSAC to determine the position and pose of the screw, respectively. For higher accuracy, KF is introduced to correct the biased positioning and pose estimation results.

The latest one-stage detection network YOLOv5 is selected to perform the screw detection tasks owing to the YOLO network’s high accuracy and fast reasoning speed. The network takes the image captured by the camera as input and treats the center (x_c, y_c) of the detected screw’s bounding box as the center of its upper surface. The detection model is trained in a supervised process on a set of 1200 labeled images, simultaneously using data augmentation to extend the number of different images. The dataset contains two categories: hex screw and hex nut, wherein the hex screw account for the 80% of the total number of fasteners. As for the three models of YOLOv5 network, YOLOv5-S stands out after our investigations with $AP_{75}97.78\%$. We train YOLOv5 by learning rate 1×10^{-4}, batch size 10, input image size $416 \times 416 \times 3$. The training process lasts for 100 epochs.

After we get the bounding box of the screw in the RGB image through YOLOv5, combined with the camera position and depth information, we can get the spatial coordinates of each point on the screw. In theory, as long as we have the spatial coordinates of three points, we can calculate the orientation of the screw. However, the orientation calculated in this way is almost unusable because of noise in the depth image. So we use the RANSAC algorithm to calculate the plane and the normal vector with the selected point cloud. In our case, because deformations of waste EVBs are quite common, the pose of the target screw on the edges of the upper cover is consistent with that of the cover beneath it rather than the largest plane of the vehicle battery. Given the bounding box of the screw in the RGB image, the point cloud of its adjacent areas is generated based on an expanded bounding box in the depth image. To distinguish the adjacent areas around the target screw for plane fitting, extra operations, including threshold and morphological filtering, are essential to segment the target screw from adjacent areas (upper cover) completely and efficiently.

To avoid the error of the pose estimation result caused by inaccurate centering and unreliable depth information, which is common in practice, a loop positioning method based on Kalman filtering is proposed. Kalman filtering is a highly efficient recursive filter, which is widely used in target tracking. Since the true pose of the target screw in the world frame is constant during the positioning process, Kalman filtering can be introduced to correct the estimated screw poses with Gaussian noises. In the filter, the estimated screw pose $\hat{x}$ is expressed by the center coordinates $\hat{x}^p$ and normal vector $\hat{x}^o$ of the upper surface:

$$\hat{x} = (x^p, y^p, z^p, x^o, y^o, z^o) \tag{1}$$

For the k^{th} iteration,

$$\hat{x_k}' = \hat{x}_k + K'(z_k - H_k\hat{x}_k) \tag{2}$$

$$P_k' = P_k + K'H_kP_k \tag{3}$$

$$K = H_kP_kH_k^T(H_kP_kH_k^T + R_k)^{-1} \tag{4}$$

where P_k is the covariance matrix, H_k is the identity matrix, R is the covariance matrix (representing the sensor noise), and P_0 is the initial covariance estimation matrix.

After each iteration, the end effector will be adjusted to be consistent with the filtered screw posture and start the following recognition just above the screw to reduce the positioning error caused by the tilt of the screw in the image. Besides calculating the position, the Kalman Filter can predict the uncertainty (variance) of the calculated position. We use it as a stop condition. When the variance is less than the pre-set threshold, the filtering process stops and uses the result for the next primitive.

C. Insertion Based on Force Perception

Besides minimizing the visual positioning error through the Kalman filter, we rely on a force perception-based Markov decision process model to ensure high robustness via combining active and passive compliance in the end effector. The contact state criterion and arc trajectory search algorithm based on force perception are introduced to establish a Markov decision process (MDP) model (as shown in Figure 3). The current contact state is judged by the force feedback transmitted at the end effector sleeve with high sensitivity, and then the arc trajectory search corresponding to different radii or the reinserting attempt is performed until the sleeve is successfully clamped.

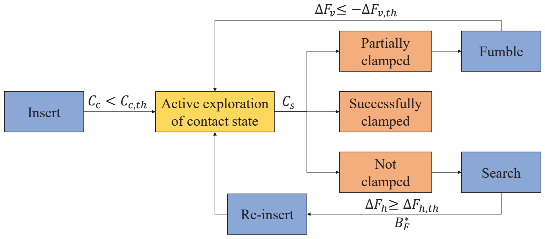


Fig. 3. MDP model based on force perception

Assume that there are three types of states when the sleeve is in contact with the screw or battery pack (as shown in Figure 4):

1) Successfully clamped ($s_t = success$): the inner wall of the sleeve completely surrounds the side of the screw, which is regarded as a complete clamping.
2) Partially clamped ($s_t = part$): the lower edge of the sleeve is in contact with the upper surface of the screw and is thus under relatively larger pressure, which is regarded as a partial clamping.
3) Not clamped ($s_t = lost$): the sleeve is in direct contact with the surface of the battery pack, which is regarded as totally not clamped.

In this way, a discrete sequence $s(n)$ can be used to describe the dynamic change of the contact state:

$$s(t) = (s_0, s_1, \ldots, s_t) \tag{5}$$

where $s_0 = init$ is the initial state. Each action decision a_t of the MDP model is made based on the current state s_t, whose purpose is to finally transform the contact state into the goal state $s_G = success$.

In order to characterize the current contact state, we use a sliding time window of length n to obtain the force feedback sequence $F(n)$ of the most recent moments:

$$F(n) = (F_{t+1-n}, F_{t+1-(n-1)}, \ldots, F_t) \tag{6}$$

The advantage of describing features in the form of sequences is that it can prevent misjudgments caused by delays or errors in force feedback at certain moments. When $F(n)$ conforms to certain characteristics, the active exploration of the contact state is applied to further determine the specific contact state, that is, it makes slight translations in its

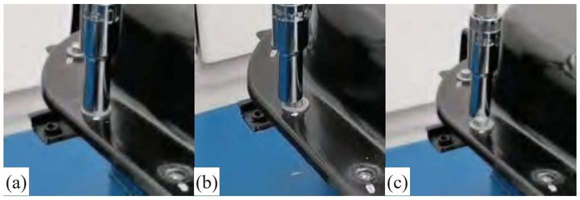


Fig. 4. Different contact states of the screw and the sleeve: a) Successfully clamped. b) Partially clamped. c) Not clamped.

horizontal directions, which includes two adjacent moves in both positive and negative direction of each axis (to prevent misjudgment caused by collision with surrounding objects when clamping is failed).

In this way, the model can be described by the following framework:

$$s_{t-1} = init : a_{t-1} = insert \rightarrow s_t$$

$$s_t = \begin{cases} init, & C_c \geq C_{c,th} \\ success, & C_c < C_{c,th} \text{ and } C_s < C_{s,th} \\ part, & C_c < C_{c,th} \text{ and } C_s \geq C_{s,th} \end{cases} \quad (7)$$

When executing the "Insert" primitive, the robot first moves the sleeve a certain distance above the estimated screw position (in case of premature collision due to error in estimation) and record the initial force F_0. The contact state starts with $s_0 = init$, and the corresponding action is to descend while rotating in constant tiny steps with force feedback F_i. The contact coefficient C_c with the threshold $C_{c,th}$ is defined to judge whether the sleeve is in contact with the screw or the surface of the battery pack:

$$C_c = \sum_{i=t+1-n}^{t} ||(F_{z,i}, F^*_{z,i})||^2_{\sigma_{C_c}} \quad (8)$$

where $\{F_{z,i}\}$ is the measured force of the sleeve minus F_0 in its vertical direction, and $\left\{F^*_{z,i}\right\}$ is the estimated force after the sleeve begins to contact the screw, and $||\cdot||_\sigma$ is the Euclidean distance with Gaussian noise of standard deviation σ. When $C_c < C_{c,th}$, it is considered that the screw has descended to the depth of successful clamping and touched the surface of the battery pack.

Since the accuracy of the above-mentioned pose estimation method based on visual perception is within 5 mm, when the first contact occur, the possible contact states are only successfully or partially clamped. In this case, the successful-clamping coefficient C_s is defined to judge whether the sleeve is successfully clamped:

$$C_s = \sum_{i=t+1-n}^{t} \left(||(F_{x,i}, F^*_{x,i})||^2_{\sigma_{C_{s,x}}} + ||(F_{y,i}, F^*_{y,i})||^2_{\sigma_{C_{s,y}}} \right) \quad (9)$$

When C_s is less than a certain threshold $C_{s,th}$, the sleeve is considered successfully clamped.

$$s_{t-1} = part : a_{t-1} = fumble \rightarrow s_t$$

$$s_t = \begin{cases} part, & \\ success, & \Delta F_v \leq -\Delta F_{v,th} \text{ and } C_s < Cs,th \\ lost, & \Delta F_v \leq -\Delta F_{v,th} \text{ and } C_s \geq Cs,th \end{cases} \quad (10)$$

Otherwise, when partially clamped, the "Fumble" primitive is performed, whose principle is to namely imitate human behavior of fumbling around to achieve clamping by short-step circular arc trajectory search with a gradually increasing radius. In this process, as long as the force sensor detects a sudden reduction $\Delta F_z > -\Delta F_{z,th}$ in the vertical force feedback, the active exploration should be executed again. This time, if $C_s \geq C_{s,th}$, it is considered that the sleeve has lost contact with the screw, and the status transitions to not clamped.

$$s_{t-1} = lost : a_{t-1} = search \rightarrow s_t$$

$$s_t = \begin{cases} lost, & \Delta F_h < \Delta F_{h,th} \\ success, & \Delta F_h \geq \Delta F_{h,th} \text{ and } C_s < Cs,th \\ part, & \Delta F_h \geq \Delta F_{h,th} \text{ and } C_s \geq Cs,th \end{cases} \quad (11)$$

When not clamped, the "Search" primitive is executed. A circular arc trajectory search with a longer step and a gradually increasing radius will be performed in this process to search for the screw and make contact with it. Once the force sensor detects a sudden increase $\Delta F_h \geq \Delta F_{h,th}$in the horizontal force, the repositioning of the screw based on force perception is achieved:

$$B^*_F = \begin{bmatrix} R_S & t_S - (r_S + r_B)t_F \\ 0 & 1 \end{bmatrix} \quad (12)$$

where B^*_F is the SE(3) representation of the estimated screw pose, R_S and t_S are from the SE(3) representation of the sleeve pose, r_S and r_B are the outer radii of the sleeve and screw, respectively, and t_F is the normalized direction vector of the force on the sleeve.

The updated pose estimation B^*_F serves as the input of the "Re-insert" primitive, which lifts the nut runner up and descends to the target location trying to insert again, followed by another turn of active exploration. On this basis, the decision model will keep making attempts to insert until the screw is finally successfully clamped.

IV. Experiments and Results

To test the robustness of this NeuroSymbolic TAMP for the battery disassembly task, we conducted a series of tests with and without obstacles on a self-built platform. We did an ablation study for the vision-based screw pose estimation. We also evaluated the overall success rate of insertion and the accuracy of re-positioning with the force-sensing-based MDP model. The test results show that the method proposed in this paper can ensure high robustness.

A. Experimental Configurations

Figure 5 shows the experiment platform, on which a used vehicle battery pack and a 6-DOF robot UR10e, a product of UR, are placed. The robot arm is equipped with the customized nut-runner at its end. PDDL-based NeuroSymbolic TAMP control the system to finish different screw disassembly tasks based on a force torque sensor and a depth camera. The torque sensor is built in the last joint of UR robot and the depth camera is Intel RealSense D435 and mounted on the end effector.

B. Vision-based Pose Estimation and Neural Predicates

To verify the accuracy of the pose estimation, the system randomly selects the screws on the battery pack and performs 120 estimations with each method. What's more, we randomly select half of them and set up obstacles next to the screws to verify the capability of Neural Predicates. Neurosymbolic TAMP will control the robot to finish the screw disassembly task.

To understand each module's contribution to the system's success rate more clearly, we repeat the experiments of

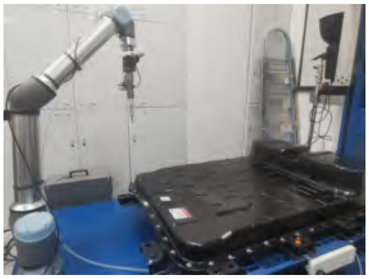


Fig. 5. Work platform equipped with robot arm

position estimation in 4 configurations. In the ’YOLO only’ configuration, the system detect the screw position in the RGB image through YOLOv5. Then the system finds 3 points on the screw surface, calculates their spatial coordinates based on the depth information, and obtains the screw angle finally. In the ’YOLO w/ KF’ configuration, the system introduces the Kalman Filter based on ’YOLO only’ and figures out more accurate positions through multiple observations. In the ’YOLO w/ RANSAC’ configuration, the system introduces RANSAC to calculate the orientation of the screw. In YOLO w/ KF & RANSAC configuration, the system uses the Kalman Filter and RANSAC algorithm at the same time. The errors of pose estimation and the success rates of subsequent insertion are shown in Table I. Given that the pose repeatability of a UR10e is ± 0.05 mm, the error caused by the execution of the robot is ignored in this experiment.

The results show that RANSAC algorithm can help to provide accurate estimated orientation and improve the success rate significantly. At the same time, Kalman Filter can help to get more accurate postion and orientation estimation and help to improve the success rate to 100%. In summary, the proposed system can satisfy the restrict requirement of EVBs disassembly. Besides, a small positioning error also avoids serious misalignment that could invalidate the contact state criteria implemented in insertion stage.

In experiments, neural predicates also performed well. The neural predicate ***target_clear()*** can correctly distinguish the situation with or without obstacles every time and execute the ”Push” primitive to clear the obstacles as appropriate. The neural predicate ***target_aim()*** can help the NeuroSymbolic TAMP to trigger the ”Mate” privimive to estimation accurate position and mate the screw and sleeve if needed.

C. *Insertion Based on Force Perception*

In the above experiments, our system successfully passed all experiments relying only on visual information. However, in order to meet the stringent requirements of robustness in the industry, our system also includes designs based on force perception. In this section, we will evaluate these features. To evaluate the effectiveness and robustness of the MDP model of insertion based on force perception as well as the accuracy of active exploration proposed in this paper, a random position within 2.5-9.5 mm from the real position of the screw serves as the estimated screw position (to prevent being directly successfully clamped or too far beyond the achieved positioning accuracy to be meaningful), and 40 insertion experiments were performed using this method.

TABLE I

ACCURACY AND AVERAGE ERROR OF VISION-BASED POSE ESTIMATION

	Success Rate	Average Error			
		Position (mm)			Orientation (rad)
		x	y	z	
YOLO w/ KF & RANSAC	100%	0.38	0.37	0.55	0.017
YOLO w/ RANSAC	98.3%	0.48	0.42	0.56	0.028
YOLO w/ KF	25.8%	0.48	0.59	0.58	0.372
YOLO only	12.1%	0.55	0.61	0.61	0.428

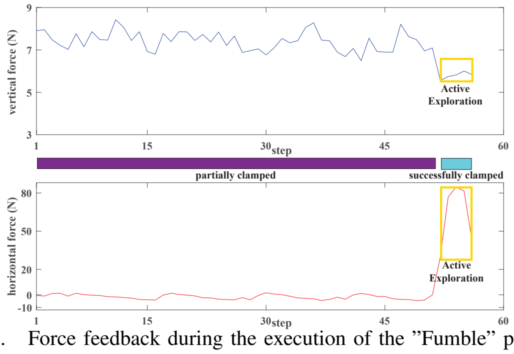


Fig. 6. Force feedback during the execution of the ”Fumble” primitive

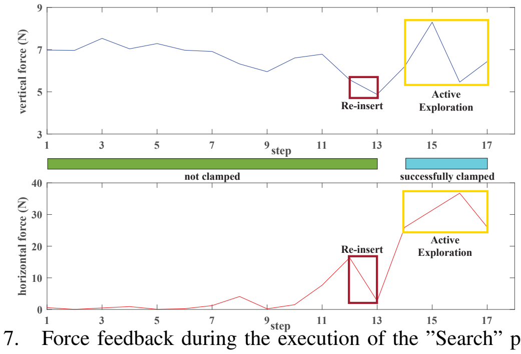


Fig. 7. Force feedback during the execution of the ”Search” primitive

Figure 6 and 7 demonstrate the force feedback during two typical processes of change in contact state dominated by the MDP model. It can be found that ideally, whether the initial state is ***part*** or ***lost***, it can easily transition to ***success*** (target state). But in practice, the active exploration of the contact state sometimes leads to unreliable judgment, which may result in redundant operations.

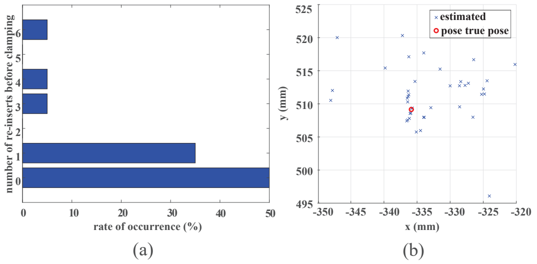
To assess the impact of such misjudgment, the confusion matrix for active exploration of contact states is obtained. From Table II, it can be found that the overall accuracy of the judgment is relatively high, and misjudgments that are prone to occur include judging ***success*** as ***part*** or judging ***part*** as ***lost***. Fortunately, if the state ***success*** is judged to be ***part***, the ”Fumble” primitive will be executed. As long as a force mutation is detected, the active search will be performed again. Similarly, if the state ***part*** is judged to be ***lost***, the ”Search” primitive will be executed, and it will generally transition to the state ***lost***.

Therefore, successful clamping is finally achieved in all the 40 experiments carried out, and the number of ”Re-insert” operation required is shown in Figure 8(a). In most cases, insertion can be completed simply by executing the ”Fumble” primitive. In the rest cases, basically only one ”Re-

TABLE II

CONFUSION MATRIX FOR ACTIVE EXPLORATION

Contact State		Predicted		
		success	part	lost
Actual	success	40	13	2
	part	0	31	16
	lost	0	1	41



(a) (b)

Fig. 8. Performance of ”Re-insert” operations: (a) attempts of ”Re-insert” required for partially clamped conditions, (b) the distribution of estimated poses based on force perception.

insert” operation is required to achieve successful clamping. In the few cases where multiple ”Re-insert” operations are required, the reason is that the screw positioning based on force perception is not accurate enough (shown in Figure 8(b)), in which further improvements can be made in the future.

V. CONCLUSIONS

To solve the problem of disassembling screws on EVBs by robot, we designed a PDDL-based NeuroSymbolic TAMP. Introducing neural predicates, the system manages to autonomously arrange the action primitives derived from manual operations to generate explainable task planning. In order to improve the robustness of the system, a customized end effector was adopted to provide passive compliance. When implementing the primitives, vision-based pose estimation with KF & RANSAC and force-based insertion we developed also contribute to the success rate. Through experiments, we verified that the pose estimation error of the system is small enough to ensure 100% success in disassembly. For large errors, it can also be compensated by force-based fumbling and repositioning. However, it should be pointed out that although the final result is not affected, the accuracy of repositioning and the judgment of the contact state can be further optimized. In the future, we will work on autonomous replacement of the sleeve on the end effector for different types of screws in EVB disassembly.

ACKNOWLEDGMENT

The authors express their sincerest thanks to the Ministry of Industry and Information Technology of China for financing this research within the program ”2021 High Quality Development Project (TC210H02C)”.

REFERENCES

[1] K. Meng, G. Xu, X. Peng, K. Youcef-Toumi, and J. Li, “Intelligent disassembly of electric-vehicle batteries: a forward-looking overview,” *Resources, Conservation and Recycling*, vol. 182, p. 106207, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921344922000556

[2] R. Li, D. T. Pham, J. Huang, Y. Tan, M. Qu, Y. Wang, M. Kerin, K. Jiang, S. Su, C. Ji, Q. Liu, and Z. Zhou, “Unfastening of hexagonal headed screws by a collaborative robot,” *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 3, pp. 1455–1468, 2020.

[3] K. Wegener, S. Andrew, A. Raatz, K. Dröder, and C. Herrmann, “Disassembly of electric vehicle batteries using the example of the audi q5 hybrid system,” *Procedia CIRP*, vol. 23, pp. 155–160, 2014, 5th CATS 2014 - CIRP Conference on Assembly Technologies and Systems.

[4] H. Chen and J. Shen, “A degradation-based sorting method for lithium-ion battery reuse,” *PLoS ONE*, vol. 12, 2017.

[5] B. Scrosati, J. Garche, and Y.-K. Sun, “20 - recycling lithium batteries,” in *Advances in Battery Technologies for Electric Vehicles*, ser. Woodhead Publishing Series in Energy, B. Scrosati, J. Garche, and W. Tillmetz, Eds. Woodhead Publishing, 2015, pp. 503–516.

[6] J. Borràs, R. Heudorfer, S. Rader, P. Kaiser, and T. Asfour, “The kit swiss knife gripper for disassembly tasks: A multi-functional gripper for bimanual manipulation with a single arm,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4590–4597.

[7] C. Klas, F. Hundhausen, J. Gao, C. R. G. Dreher, S. Reither, Y. Zhou, and T. Asfour, “The kit gripper: A multi-functional gripper for disassembly tasks,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 715–721.

[8] N. M. DiFilippo and M. K. Jouaneh, “A system combining force and vision sensing for automated screw removal on laptops,” *IEEE Transactions on Automation Science and Engineering*, vol. 15, no. 2, pp. 887–895, 2018.

[9] D. Ma, S. Dong, and A. Rodriguez, “Extrinsic contact sensing with relative-motion tracking from distributed tactile measurements,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 11 262–11 268.

[10] M. Adjigble, N. Marturi, V. Ortenzi, V. Rajasekaran, P. Corke, and R. Stolkin, “Model-free and learning-free grasping by local contact moment matching,” in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 2933–2940.

[11] A. Rastegarpanah, H. C. Gonzalez, and R. Stolkin, “Semi-autonomous behaviour tree-based framework for sorting electric vehicle batteries components,” *Robotics*, vol. 10, no. 2, 2021.

[12] A. Billard and D. Kragic, “Trends and challenges in robot manipulation,” *Science*, vol. 364, no. 6446, p. eaat8414, 2019.

[13] H. Poschmann, H. Brüggemann, and D. Goldmann, “Fostering end-of-life utilization by information-driven robotic disassembly,” *Procedia CIRP*, vol. 98, pp. 282–287, 2021, the 28th CIRP Conference on Life Cycle Engineering, March 10 – 12, 2021, Jaipur, India.

[14] X. Li, M. Li, Y. Wu, D. Zhou, T. Liu, F. Hao, J. Yue, and Q. Ma, “Accurate screw detection method based on faster r-cnn and rotation edge similarity for automatic screw disassembly,” *International Journal of Computer Integrated Manufacturing*, vol. 34, no. 11, pp. 1177–1195, 2021. [Online]. Available: https://doi.org/10.1080/0951192X.2021.1963476

[15] M. Choux, E. Marti Bigorra, and I. Tyapin, “Task planner for robotic disassembly of electric vehicle battery pack,” *Metals*, vol. 11, no. 3, 2021.

[16] J. Mahler, M. Matl, V. Satish, M. Danielczuk, B. DeRose, S. McKinley, and K. Goldberg, “Learning ambidextrous robot grasping policies,” *Science Robotics*, vol. 4, no. 26, p. eaau4984, 2019.

[17] D. Kalashnikov, A. Irpan, P. Pastor, J. Ibarz, A. Herzog, E. Jang, D. Quillen, E. Holly, M. Kalakrishnan, V. Vanhoucke, and S. Levine, “Qt-opt: Scalable deep reinforcement learning for vision-based robotic manipulation,” 2018.

[18] O. M. Andrychowicz, B. Baker, M. Chociej, R. Józefowicz, B. McGrew, J. Pachocki, A. Petron, M. Plappert, G. Powell, A. Ray, J. Schneider, S. Sidor, J. Tobin, P. Welinder, L. Weng, and W. Zaremba, “Learning dexterous in-hand manipulation,” *The International Journal of Robotics Research*, vol. 39, no. 1, pp. 3–20, 2020. [Online]. Available: https://doi.org/10.1177/0278364919887447

[19] M. Ghallab, C. Knoblock, D. Wilkins, A. Barrett, D. Christianson, M. Friedman, C. Kwok, K. Golden, S. Penberthy, D. Smith, Y. Sun, and D. Weld, “Pddl - the planning domain definition language,” 08 1998.

[20] M. Fox and D. Long, “PDDL2.1: An extension to PDDL for expressing temporal planning domains,” *Journal of Artificial Intelligence Research*, vol. 20, pp. 61–124, dec 2003.

[21] H. L. S. Younes and M. L. Littman, “Ppddl 1 . 0 : An extension to pddl for expressing planning domains with probabilistic effects,” 2004.

[22] H. Zhang, H. Yang, H. Wang, Z. Wang, S. Zhang, and M. Chen, “Autonomous electric vehicle battery disassembly based on neurosymbolic computing,” in *Intelligent Systems and Applications*, K. Arai, Ed. Cham: Springer International Publishing, 2023, pp. 443–457.